

öS 60,–/sfr 7,–/Lit 6.700/hfl 8,25 DM 7,–
AMIGA
Markt&Technik
8/9—'87 DAS COMPUTERMAGAZIN FÜR AMIGA - FANS
Kaufhilfe:
Welcher Amiga ist der beste?
Die neuesten Programme im Test
Große Marktübersicht
Software für alle
Neuer Kurs: Amiga-Basic für Einsteiger
Super-Listings zum Abtippen ★ Tips & Tricks zu Programmiersprachen
1986 Sales
Amiga-Magazin
Annual Snowfall

EPSON. Der Unterschied.



Es war gar nicht so leicht, EPSON Qualität zu so günstigem Preis im neuen LX-800 zu verwirklichen.

Jetzt kann sich wirklich jeder EPSON Qualität leisten. Dafür sorgt unser neuer LX-Drucker mit seinem günstigen Preis. Bei erstaunlich niedrigem Geräuschpegel erreicht er ein Top-Tempo von 180 Z./Sek. Und auch in Schönschrift (NLQ) bewältigt er einen für seine Klasse extrem hohen Textdurchsatz. Den Unterschied macht aber auch die Ausstattung deutlich: Zwei Schönschriften, vielfältige Schriftvarianten, volle Grafikfähigkeit, übersichtliches Bedienerfeld, Endlospapier- und Einzelblattverarbeitung. Und auf Wunsch ein vollautomatischer Einzelblatt-Einzug. Über die parallele Schnittstelle läßt sich der LX-800 an alle gängigen PCs anschließen. Für Commodore-Anwender gibt es den LX-800 mit zusätzlich eingebauter C64/128 Schnittstelle. Er ist also ein ideales Aufsteiger-Modell. Und der Preis macht das Einsteigen leicht. Fragen Sie Ihren EPSON Fachhändler.





Technologie, die Zeichen setzt.

EPSON Deutschland GmbH · Zülpicher Straße 6 · 4000 Düsseldorf 11 · Telefon 0211/5603-0

M. L. & S.

Inhalt



Welcher Amiga ist der richtige?

Ein Käufer, der sich für den Amiga entscheidet, hat leider die Qual der Wahl, welches der insgesamt drei Modelle denn nun gekauft werden soll. Soll es der »kleine« Amiga 500, der verbreitete Amiga 1000 oder das Profigerät Amiga 2000 sein? Wir sind dieser Frage auf den Grund gegangen und haben für Sie Informationen zusammengestellt, anhand deren Sie sich das für Sie am besten geeignete Gerät heraussuchen können. Gemeinsamkeiten und Unterschiede werden ebenso erwähnt wie auch geplante Erweiterungsmöglichkeiten der Systeme oder Zusatzgeräte, die bereits jetzt für den Anschluß an die Amiga-Modelle zur Verfügung stehen.

Seite 20

Farbig drucken mit dem Amiga

Der Amiga ist ein Computer mit herausragenden Farbgrafik-Fähigkeiten. Doch was bringt ein Computer, wenn die Peripherie mit diesen Möglichkeiten nichts anfangen kann? Oft wird der Wunsch nach Druckern laut, die auch die Farben des Amia aufs Papier bringen können. Wir stellen Ihnen zu diesem Zweck einige Farbdrucker vor, die farbige Ausdrucke auf Papier ermöglichen.

Seite 116



AKTUELLES

Bericht von der Comdex: Amiga trumpft auf — 9
Neue Produkte — 11
Interview mit Atlantis: Neue Wege gehen — 13
Kurztest: dbMAN — 14
Amiga im Unterricht — 15

EINSTEIGER

Kaufhilfe: Welcher Amiga ist der richtige? — 20
Frühjahrsputz für die Workbench-Disketten — 28
Literatur für Einsteiger — 29
Neu: AMIGA GLOSSAR (AB—CO) — 67

SOFTWARE-TEST

Die neuesten Programme im Test
Diskettendateien im Griff (Amiga Test) — 30
Durchs Prisma geschaut (Amiga Test) — 32
Assembler im Wettbewerb (Amiga Test) — 34
Komfort im CLI (Amiga Test) — 106
Dr. Hacker zum OP bitte! (Amiga Test) — 107
Die Tastatur im Griff mit »Zing!Keys« (Amiga Test) — 109
Aegis Impact (Amiga Test) — 110
Billigspiele: Preiswerter geht's kaum (Amiga Test) — 126

KURSE

CLI — Die zweite Haut des Amiga (Teil 2) — 45
Neuer Kurs: Basic für Aufsteiger (Teil 1) — 49

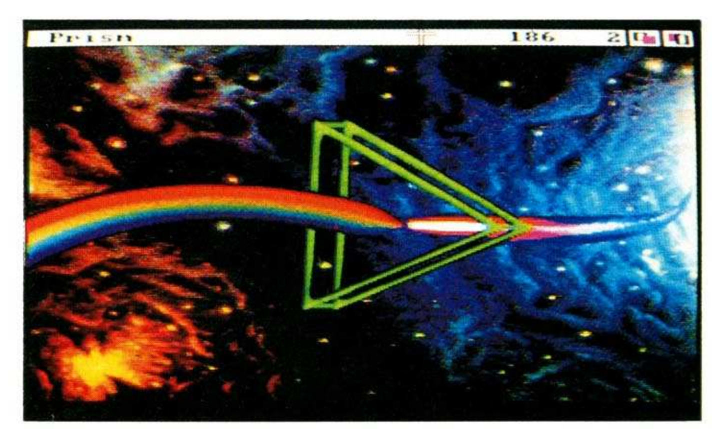
LISTINGS ZUM ABTIPPEN

Programm des Monats: CLI-Befehle mit der Maus — 37
Brücken schlagen — mit Convert 64 — 53
Veränderung des CLI — 60
Flexibles Wörterbuch — 64
Biorhythmus mit Extras — 74
Vier gewinnt — 77
Hilfsmittel zur Eingabemasken-Erstellung — 82

TIPS & TRICKS

Tips & Tricks zum Amiga — 86
Datenübertragung mit dem Amiga — 89

8/9 1987



Software unter der Lupe

Wie immer haben wir natürlich auch in dieser Ausgabe wieder einige Bonbons unter den Software-Neuerscheinungen der letzten Wochen für Sie getestet. Zum Beispiel »Prism«, das brandneue Malprogramm für den Hold-and-Modify-Modus des Amiga. Jetzt können endlich auch H.A.M-Bilder mit Komfort nachbearbeitet werden. Außerdem beschäftigen wir uns ausführlich mit »Aegis Impact«, einem Kalkulationsprogramm mit grafischer Werteausgabe. Ebenfalls dabei: »Surgeon«, ein Programm, mit dem es möglich sein soll, am heimischen Bildschirm Operationen wie im Krankenhaus zu simulieren. **ab Seite 32**

Diskettendateien im Griff

Auf der Workbench-Ebene können nicht alle Dateien auf einer Diskette beeinflußt werden. Dies ist nur über das CLI (Command-Line-Interface) möglich. Will man aber von CLI aus viele Dateioperationen ausführen, kann dies leicht in wüste Tipparbeit ausarten. Abhilfe schafft das Programm »CLI-Mate«, das die Tipparbeit auf ein Minimum beschränkt, da das Programm voll mausgesteuert arbeitet und eine Vielzahl von nützlichen Funktionen bereitstellt. Außerdem ist CLI-Mate eine große Erleichterung für Amiga-Benutzer, die nur über ein Laufwerk verfügen.

Seite 30



PROGRAMMIEREN

Simple Sprites in C	91
Hardcopy perfekt	94
Speicheranzeige im CLI	96

SOFTWARE

Große Marktübersicht:

Software für alle Amiga-Modelle	98
Die Public Domain-Seite	104

HARDWARE

Schneller durch mehr Speicher	112

HARDWARE-TEST

Theorie und Praxis des Druckens mit dem Amiga	116
1 MByte RAM für den Amiga 1000	122
Aus RGB wird PAL	123
Genlock-Interface: Die Trickkiste	124

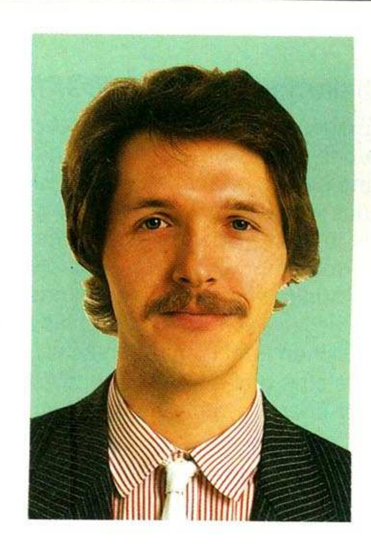
AUFRUFE UND WETTBEWERBE

Gewinnen Sie 2000 Mark...	128
Ihr Programm im Amiga-Magazin	129
Gesucht: Tips & Tricks zum Amiga	129

RUBRIKEN

Editorial	8
Leserforum	16
Programmservice	25
Guru Meditation	90
Bücher	127
Vorschau	131
Impressum	131

Dieses Symbol zeigt an, welche Programme auf Diskette erhältlich sind.



Hilfestellung

Sie erhalten in dieser Ausgabe eine vollständige Liste aller Programme, die es für die Amiga-Familie bereits zu kaufen gibt, oder die demnächst zu erwerben sind. Doch täglich reihen sich neue, faszinierende Programme ein. Und fast täglich bewegen sich auch die Preise in angenehmere Regionen — das heißt nach unten. So wurden während der Zusammenstellung unserer Marktübersicht fast jede Woche neue Preise von den Anbietern und Vertreibern bekanntgegeben, die teilweise drastisch unter denen der Vorwoche lagen. Zum Zeitpunkt des Erstverkaufstags des Amiga-Magazins werden mit Sicherheit einige Programme preiswerter erhältlich sein, als in unserer Übersicht angegeben.

Neben der zunehmenden Menge an Programmen ist gerade der Trend der sinkenden Preise für den Kunden als erfreulich zu verzeichnen. Welche Auswirkungen ergeben sich daraus?

Zum einen ist sicherlich der Softwaremarkt für den Amiga gerade am Erblühen, und die Anbieter beziehungsweise Hersteller versuchen einen möglichst großen Teil des Kuchens für sich zu gewinnen. Jeder Produzent oder dessen Distributor will selbstverständlich die Startlöcher optimal besetzen, um bei der zu erwartenden Hausse vorne mit dabeizusein.

Das bedeutet, Sie als Kunde bekommen gute Software, in die die Programmierer sehr viel Arbeit und Energie gesteckt haben, für einen vernünftigen, monetären Gegenwert. Die ausführliche, didaktisch gut aufbereitete Anleitung ist in diesem Preis sogar noch mit dabei. Was aber bei diesen Preisentwicklungen auf der Strecke bleiben muß — rein aus kaufmännischen Überlegungen heraus — ist der Support, die Hilfestellung nach dem Kauf. Der erfahrene Computeranwender mag sich damit zufriedengeben, kennt er doch alle oder die meisten Programme aus seinem Spezialgebiet. Auch findet er sich aufgrund seines Know-hows sehr schnell in anderen Bereichen zurecht.

Der Fortgeschrittene oder Einsteiger fühlt sich ziemlich alleingelassen.

Hier greift das Amiga-Magazin ein. Sie sollen sich durch unsere Testberichte zunächst einmal für das richtige Produkt entscheiden. Auch nach dem Kauf werden wir Sie mit nützlichen Tips und Tricks zum Programm, mit Informationen, wie andere »User« bestimmte Probleme gelöst haben, überhäufen. Dies bedingt natürlich, daß Sie sich rege an der Informationssammlung beteiligen, das heißt Ihre Lösung allen »Leidensgenossen« via Zeitschrift zukommen lassen. Wenn der Support nicht automatisch erfolgt, muß er über andere Kanäle laufen. Wir sind bereit dazu, und Sie?

Ihr
Albert Absmeier,
Chefredakteur

68000er

Buchstäblich in letzter Sekunde gelang es uns, einen topaktuellen Bericht über die amerikanische »Comdex«-Messe in unsere **68000er-Ausgabe 8/87** zu packen. Lesen Sie, welche brandheißen Neuheiten für den Amiga dort erstmals öffentlich präsentiert wurden.

Licht in den Programmiersprachen-Dschungel bringt unsere ausführliche Übersicht der wichtigsten Sprachen von Assembler über C bis Basic.

Leichte Bedienung, Archivieren von Grafiken und Sound-Dateien und eine hohe Geschwindigkeit: Das sind die Hauptmerkmale der neuen Dateiverwaltung »GoAmiga-Datei«, die ihre Leistungsfähigkeit in unserem Test unter Beweis stellt.

Wegen des riesigen Erfolgs unseres 68000er-Sound-Digitizers haben wir die Lötkolben abermals angeheizt und die Schaltung um ein paar Bauteile erweitert, damit der Digitizer auch mit dem Senkrechtstarter Amiga 500 einwandfrei zusammenarbeitet.

Zudem sind in der August-Ausgabe wieder viele News, Grundlagen und Listings zum Abtippen dabei.

Das Thema Grafik schreiben wir in der **Ausgabe 9/87** des 68000er-Magazins besonders groß. Wer ist besser? Amiga-Blitter oder Atari ST-Blitter? Lesen Sie, was der Atari ST-Blitter alles kann.

Ein Meister aller Grafik-Klassen ist Dale Luck. Als Mitglied der legendären Amiga-Entwickler-Crew kennt er die ausgefeilte Technik und die interessante Geschichte dieses faszinierenden Computers wie kein anderer. Lernen Sie diese schillernde Persönlichkeit mit uns kennen.

Telekommunikation muß nicht trocken sein: Diga! heißt der neue Champion bei den Amiga-Terminalprogrammen. Ob Zeichensatz, Auflösung, Terminalemulationen oder Übertragungs-Protokolle: alles ist bei Diga frei wählbar.

Jetzt sind endlich alle CLI-Befehle, DOS-Fehlermeldungen und Editor-Kommandos zentral zusammengefaßt: Unsere Amiga-DOS-Referenzkarte sollte bei keinem Amiga-Besitzer fehlen. Ideal zum an die Wand hängen und Überblick behalten.

Was wir überdies noch an Überraschungen für Sie haben, wird noch nicht verraten!



Die **Ausgabe 8/87** der Happy-Computer beschäftigt sich ausführlich mit dem Thema Computer und Musik. Im musikalischen Dreikampf vergleichen wir drei Musikprogramme für den Amiga und zeigen ihre Stärken und Schwächen. Lesen Sie, welches Programm für Sie am besten ist, um die Soundfähigkeiten des Amiga voll auszuschöpfen. Bei Musik wird auch beim Amiga MIDI immer wichtiger und immer mehr Programme sind auch MIDI-fähig. Damit Sie beim Kauf eines MIDI-Keyboards nicht hereinfallen, haben wir die preiswertesten und besten getestet.

Musik macht der Amiga wie kein zweiter, doch bei 16 Millionen Farben wird selbst er neidisch. Wir zeigen, wie ein professioneller Computergrafiker die fazinierenden Trickfilme mit dem Computer erzeugt.

»StarNet« — Die Herausforderung für Programmier-Cracks. Schreiben Sie die beste Umsetzung unseres fesselnden Taktikspiels für den Amiga. Der Clou: Sie können das Spiel überall mit Freunden auch ohne Computer spielen. Das Spielbrett befindet sich in der Mitte zum Heraustrennen.

Jede Menge Messeneuheiten stellen wir in der **Ausgabe 9/87** vor, die Mitte August erscheint. Wir haben uns für Sie auf der Commodore-Show in London umgesehen und viele interessante Neuheiten — von Textverarbeitung bis CAD-Programm — für den Amiga und den C 64 entdeckt.

DFÜ wird immer beliebter. In einem Vergleichstest treten nicht nur drei DFÜ-Programme für den Amiga gegeneinander an, wir stellen Ihnen auch eine super DFÜ-Anwendung vor: Multi-User-Adventures. Viele Spieler kämpfen gleichzeitig in einem gigantischen Labyrinth.

Für alle Spiele-Freaks gibt es nicht nur den brandaktuellen Spiele-Teil mit den Tests der neuesten Amiga-Spiele, sondern auch zwei weitere heiße Eisen: Sind wirkliche alle Computerbesitzer Kriminelle, wie manche behaupten? Wir beleuchten die Raubkopiererszene von verschiedenen Seiten und bringen Licht in den Dschungel von Meinungen und Unklarheiten. Unser zweites Thema läßt alle begeisterten Spieler aufhorchen. Der Happy-Joystick zum Basteln macht schwere Spiele leichter.



Amiga trumpft auf

Auf der Comdex Spring in Atlanta war der Amiga 2000 stark vertreten. Hier konnten Subaussteller auf dem Commodore-Stand demonstrieren, welche Anwendungen mit dem Computer bereits realisiert sind, oder demnächst noch werden. Der Trend beim Amiga 2000 geht eindeutig in Richtung professionelle Applikationen wie Desktop Publishing, CAD oder Kalkulation.

Wer, wie bisher, Commodore auf der größten Messe für Unterhaltungselektronik, der Summer Consumer Electronics Show CES in Chicago vermutete, mußte Anfang Juni schleunigst umdisponieren. Das Unternehmen setzte in diesem Frühling voll auf die Präsentation des Amiga 2000 in den USA. Mit der Stoßrichtung, professionelle Anwender zu gewinnen, war die Comdex in Atlanta, Georgia, gerade richtig. Die Comdex ist eine der wichtigsten Messen für die Personal Computer und deren Software. Kleine Anekdote am Rande: Atari war nicht auf der Comdex, sondern mit einem pfiffigen Stand auf der CES in Chicago zu finden.

Commodore hatte einigen Herstellern von Soft- oder Hardware Platz auf dem eigenen Stand zur Verfügung gestellt. Diese demonstrierten teilweise bekannte Produkte, aber auch viele interessante Neuentwicklungen.

Eine Sparte für die sich der Amiga besonders gut eignet ist sicherlich Desktop Publishing. Hier werden hohe Rechnerleistungen mit schneller grafischer Ausgabe gefordert. Die erste Generation der Desktop Publishing-Programme ist bereits lieferbar, konnte allerdings den Profi mit seinen gehobenen Ansprüchen noch nicht so recht befriedigen. Oft wurden sogar »normale« Textverarbeitungsprogramme, nur weil sie Grafiken einbinden und mehrspaltigen Text drucken konnten, bereits als DTP-Programme proklamiert. Ein bekanntes DTP-Programm der ersten Stunde ist PageSetter von Gold Disk (siehe Test im Amiga-Magazin, Ausgabe 6/7, Seite 98). Von derselben Firma hat dieses Programm einen großen Bruder bekommen: Professional Page. Der Name verspricht viel, das Demoprogramm auf der Messe konnte die Erwartungen erfüllen. Es lassen sich sämtliche Grafiken im IFF-Format verwenden und in Schwarzweiß-Halbtönen ausdrucken.

Eigene Zeichnungen malt man mit einem CAD-Modul (hier als Computer Assisted Drawing bezeichnet), das erstmals direkt mit einem DTP-Programm kombiniert wurde. Dieses Textbearbeitungsprogramm kann sich zwar nicht ganz mit »reinen« Profis messen, ist aber eines der besten und schnellsten DTP-Programme. Professional Page soll 395 Dollar kosten und noch vor Weihnachten erhältlich sein.

Um Vierfarbbilder in 16 Graustufen auf dem Bildschirm und auf dem Drucker (es werden sämtliche Laserdrucker mit Postscript und sogar Linotronic-Satzmaschinen mit 2400 dpi unterstützt) zu erhalten, ist ein 100 Dollar teueres Zusatzprogramm nötig.



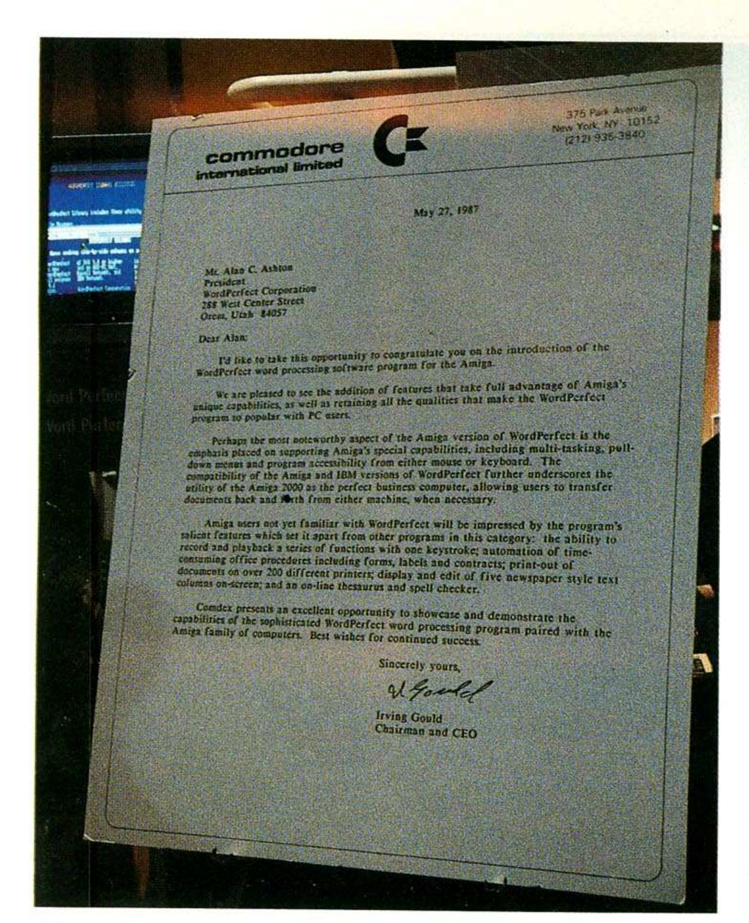
Bild 1. Die Wettervorhersage mit dem eigenen Computer schnell erstellt: »Weather Graphic« von ACS

Andere Vertreter des DTP-Bereiches, wie Publisher 1000 von N.E. Software Group für 200 Dollar oder City Desk von Microsearch für 100 Dollar, wurden auf der Messe als lieferbar bekanntgegeben.

Zum Zeitpunkt des Erscheinens dieser Ausgabe dürfte auch das aus der MS-DOS-Welt bekannte Textverarbeitungsprogramm Wordperfect von Wordperfect Corporation in der amerikanischen Version bereits über den Ladentisch gehen. Die deutsche Version soll noch zwei bis drei Monate auf sich warten lassen. Sämtliche Features, die Wordperfect für den PC zum Bestseller werden ließen, sind nach Angaben eines Firmenvertreters auch in der Amiga-Version enthalten. Die Bedienung kann entweder über die Tastatur oder über die Maus erfolgen. Das Programm ist mit 395 Dollar angekündigt worden. Sehr stolz zeigte sich Commodore über den Entschluß von Wordperfect Corporation, ihr Produkt auf den Amiga umzuschreiben. Deshalb wurde auch gleich der erste Brief, der mit diesem Programm geschrieben wurde, auf etwa 1,5 x 2 Meter vergrößert und am Stand präsentiert (Bild 2). Commodore verspricht sich durch diese Adaption eines bekannten PC-Programms Signalwirkung auf die wirklich großen Software-Hersteller wie Lotus oder Microsoft.

Neue Wege

In einem Segment, das man kaum erwähnt, wenn man die Stärken des Amiga aufzählt, hat sich ein ausgezeichnetes Produkt etabliert: Maxiplan Plus für 199 Dollar von Maxi-



commodore international limited

375 Park Avenue
New York, NY 10152
(212) 935-3840

May 27, 1987

Mr. Alan C. Ashton
President
WordPerfect Corporation
288 West Center Street
Orem, Utah 84057

Dear Alan:

I'd like to take this opportunity to congratulate you on the introduction of the WordPerfect word processing software program for the Amiga.

We are pleased to see the addition of features that take full advantage of Amiga's unique capabilities, as well as retaining all the qualities that make the WordPerfect program so popular with PC users.

Perhaps the most noteworthy aspect of the Amiga version of WordPerfect is the emphasis placed on supporting Amiga's special capabilities, including multi-tasking, pull-down menus and program accessibility from either mouse or keyboard. The compatibility of the Amiga and IBM versions of WordPerfect further underscores the utility of the Amiga 2000 as the perfect business computer, allowing users to transfer documents back and forth from either machine, when necessary.

Amiga users not yet familiar with WordPerfect will be impressed by the program's salient features which set it apart from other programs in this category: the ability to record and playback a series of functions with one keystroke; automation of time-consuming office procedures including forms, labels and contracts; print-out of documents on over 200 different printers; display and edit of five newspaper style text columns on-screen; and an on-line thesaurus and spell checker.

Comdex presents an excellent opportunity to showcase and demonstrate the capabilities of the sophisticated WordPerfect word processing program paired with the Amiga family of computers. Best wishes for continued success.

Sincerely yours,

Irving Gould
Chairman and CEO

Bild 2. Der erste Brief mit Wordperfect auf dem Amiga, geschrieben von Irving Gould, Chairman von Commodore



Bild 3. Der Printer/Plotter Colormaster von CalComp druckt in über 4000 Farbschattierungen



Bild 4. A1-Plotter für X-CAD und Dynamic-CAD

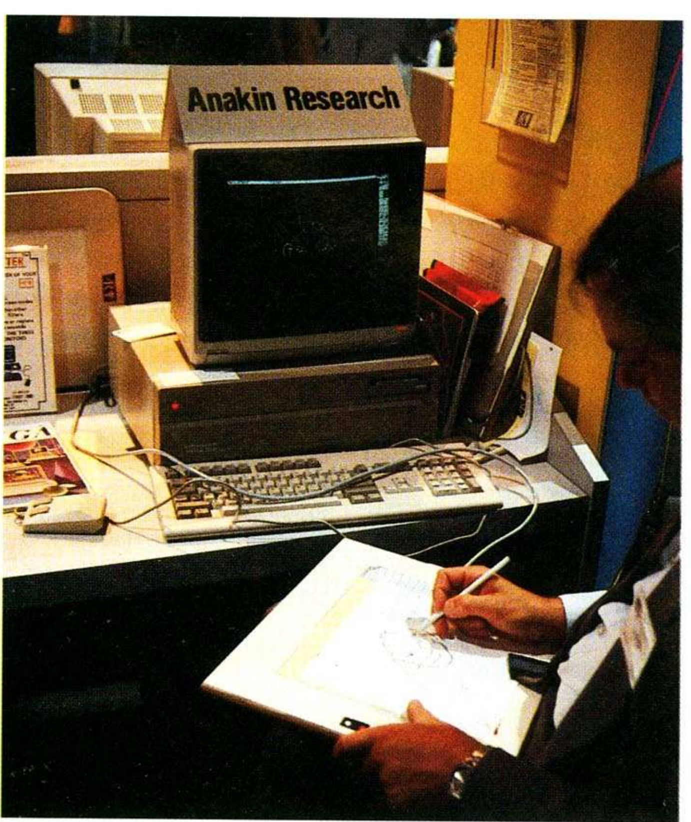


Bild 5. Einfache Eingabe von Zeichnungen und Grafiken mit dem Grafiktablett »Easyl«

soft. Dieses Tabellenkalkulationsprogramm nutzt so richtig die innewohnenden Eigenschaften des Amiga aus. So ist es beispielsweise bei ähnlichen Programmen möglich, durch Veränderung der Werte sofort die Auswirkungen in einem übersichtlichen Diagramm zu beobachten. Bei Maxiplan Plus funktioniert das aber auch in die andere Richtung, das heißt: Sie variieren mit dem Mauszeiger einen Punkt im Liniendiagramm und der korrelierende Wert sowie sämtliche damit verknüpften Formeln im Arbeitsblatt passen sich automatisch an.

Ganz andere Eigenschaften des Amiga benötigen die Grafik- und Animationsprogramme. Von NewTek gibt es für die digitalisierten Bilder des Digi-View jetzt endlich das entsprechende Malprogramm Digi-Paint für 60 Dollar. Digi-Paint ist ein Malprogramm, das alle 4096 Farben (Hold and Modify-Modus) verwenden kann. Spezielle Weichzeichner-Modi verleihen den Bildern das Aussehen eines Aquarells oder Ölgemäldes.

Mit »nur« 32 Farben, dafür aber mit ziehenden Wolken kommt »Station Manager Weather Graphics« aus (Bild 1). Dieses Programm dient der Visualisierung der Wettervorhersagen. Die Wetterdaten kommen via Satellit oder per DFÜ. Sicher eine spezialisierte Anwendung, die aber die breiten Einsatzmöglichkeiten des Amiga aufzeigt.

Die farbenprächtigen Bilder von Digi-Paint, Impact oder Prism lassen sich nach Aussagen des Anbieters optimal auf dem Colormaster (Bild 3) von CalComp ausgeben. Der Printer/Plotter arbeitet mit einer Auflösung von 200 dpi (dots per inch) und kann über 4000 Farbschattierungen wiedergeben. Über 4000 Dollar kostet der Printer/Plotter; ein farbiger Ausdruck kommt auf 25 Cents.

Ein astreiner A1-Plotter (Bild 4) ist anzuraten, wenn man die CAD-Programme X-CAD von Taurus und Dynamic-CAD von Microillusions auf dem Amiga bis an die Grenzen ausfahren will. Beide Produzenten sind sich in ihren Werbeaussagen einig: Wollte man die Leistung,

die das Programm heute bietet, vor einem Jahr haben, so mußte man mehr als 40000 Dollar dafür aufwenden. X-CAD zeichnet 3000 Linien pro Sekunde auf den Bildschirm. Für 2000 Kreise benötigt das bekannte Auto-CAD auf den PCs 2 Minuten, mit X-CAD ist die Arbeit in 12 Sekunden erledigt. Das englische Unternehmen Taurus bietet verschiedene »Workstations«, beginnend mit einem A 500 mit Monitor als Basiseinheit für 2000 Pfund Sterling, bis hin zum 8000 Pfund teurem X-Station mit MC 68020, Ethernet LAN und PC-Kompatibilität an. Dynamic-CAD kostet als Einzelprodukt 500 Dollar.

Nicht mit Maus oder Tastatur werden die Grafiken bei Anakin Research in den Amiga gezaubert, sondern mit dem druckempfindlichen Grafiktablett Easyl (Bild 5). Mit einem normalen Bleistift wird die Vorlage abgefahren und die Daten an Programme wie Deluxe Paint II oder PageSetter übermittelt. Das Tablett gibt es in Deutschland für 998 Mark bei DTM.

Commodore ist es gelungen, mit der Präsentation auf der Comdex den Amiga 2000 in die Welt der Business-Computer einzuführen. Offene Anerkennung und verstohlene, neidische Blicke der Konkurrenz wechselten sich ab. Wem es bisher nicht klar war, der hat es spätestens auf dieser Messe erfahren; der Amiga 2000 ist ein erfolgreicher Herausforderer gegen die etablierte PC-Welt. (aa)

Handy Scanner für Amiga 2000

Das kleine, aber sehr feine Gerät (Bild 1) dient zum Abscannen von Grafik- und Textvorlagen, die sich dann als monochrome Grafik im Speicher des Computers befinden. Derzeit ist der Handy Scanner zu einem Preis von 898 Mark für den IBM-PC und Kompatible erhältlich. Einem Einbau in den Amiga 2000 mit PC-Karte steht also nichts im Weg. Zum Lieferumfang gehören der Scanner selbst, ein Gerät in Zigarettenschachtelgröße, eine PC-kompatible Steckkarte, eine Diskette mit Treibersoftware und ein Handbuch. Die Software erlaubt mit dem Programm »SETUP« eine komfortable Installation der Scannersoftware. Anpassungen für die gebräuchlichsten Grafikkarten und Drucker werden mitgeliefert. Zudem befindet sich auf Diskette ein mausgesteuertes Pogramm, das das Arbeiten mit dem Scanner sehr komfortabel gestaltet. Das Format der mit dem Handy Scanner abgetasteten Bilder auf Diskette ist kompatibel zu vielen Grafik-Programmen wie etwa »Dr. Halo«, »PC-Paintbrush« oder »Starwriter«. Derzeit arbeitet Cameron an einer Version des Scanners für den Amiga 500 und 1000. Geplant ist nach Angaben des Herstellers zudem eine Mustererkennung, die es gestattet, eingelesene Textvorlagen, die ja zuerst nur als Grafik im Speicher stehen, in ASCII-Text umzuwandeln. Dieser kann dann mit einem beliebigen Textprogramm weiterverarbeitet werden. (sk/jk)

Bezugsquelle: Cameron — Elektronische Produkte GmbH, Pfaffenweg 1, 7024 Filderstadt, Handy Scanner für PC: 898 Mark



Ein Scanner zum Vorlagen-Übertragen

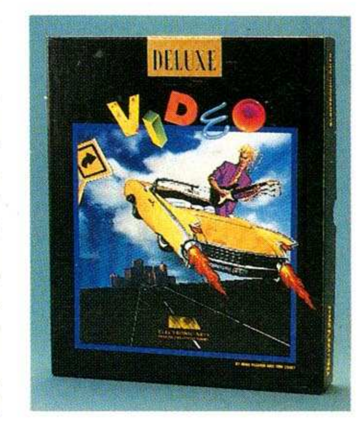
Der Klassiker in Neuauflage

Deluxe-Video von Electronic Arts liegt nun in der neuesten Version 1.2 vor.

Gleich vorweg — geändert hat sich relativ wenig. Die Funktionen und Grafiken werden immer noch mit derselben Geschwindigkeit abgearbeitet und Programmabstürze sind (leider!) auch nicht wesentlich seltener geworden. Das größte Manko ist allerdings beseitigt worden: DVideo arbeitet nun auch mit zusätzlichem Speicher, so daß auch komplexere Klänge, Bilder, Szenen und Scripts hergestellt und bearbeitet werden können. Das macht sich vor allem angenehm dadurch bemerkbar, daß nicht mehr so oft von Diskette nachgeladen werden muß, was mit der Zeit doch recht lästig werden konnte. Neue Geräusche oder Bilder sind ebenfalls nicht zu finden, so daß keine weiteren erwähnenswerten Verbesserungen zu vermerken sind. Kommen wir daher, für alle die DVideo noch nicht kennen, zu einer Kurzübersicht des Programmes. Mit DVideo lassen sich beliebige Objekte (vorher gezeichnete, aus Polygonen zusammengesetzte oder aus Bildern ausgeschnittene) vor einem Hintergrund bewegen. Das Ganze kann mit Geräuschen und Musik unterlegt werden, so daß der Eindruck eines Films entsteht. Der Aufbau des Programms erinnert stark an ein Zeitdiagramm. Es stehen verschiedene Spuren für Musik, Geräusche, Hintergrundbilder, Objekte und Spezialeffekte zur Verfügung, die alle untereinander in einer einheitlichen Zeitskala angeordnet sind. So kann man zum Beispiel angeben, daß nach 25 Sekunden eine bestimmte Melodie (die vorher mit Instant-Music, Deluxe-Music oder ähnlichem erstellt wurde) gespielt wird, und für 30 Sekunden weiterläuft, während auf einer anderen Spur definiert wird, daß nach 35 Sekunden ein Männchen von links oben nach rechts unten laufen und dabei einen Schriftzug freigeben soll. Möchte man nun noch während der Anfangsmelodie den Hintergrund aus dem »Nichts« erscheinen lassen, so braucht man nicht — wie normal üblich — alle anderen Aktionen verschieben. Man richtet sich einfach eine neue Spur (Track) ein, die natürlich auch zeitlich synchron zu den vorhergehenden abgearbeitet wird, und gibt hierauf den Befehl einen Hintergrund zu laden und in der Zeit von 25 bis 30 Sekunden einzublenden. So ist der Hintergrund schon vollkommen sichtbar, wenn das Männchen mit der Schrift kommt. Mit dem Zusatzmodul »Post Production Kit« (PPK) kann man sich für bestimmte Anwendungen, wie etwa Wetterberichte, einzelne »Tracks« vorbereiten lassen, so daß man nicht mehr die ganze Arbeit leisten muß, und bestimmte Bewegungssequenzen einfacher eingeben kann. Damit ist es dann möglich, eine komplette Präsentation der Umsatzzahlen eines Betriebes in zirka 15 Minuten zu erstellen und über den Videoausgang des Amiga auf einen Videorecorder zu kopieren. Möchte man selbsterstellte Videos Freunden zugänglich machen, so ist auch das kein Problem. Mit DVideo wird ein »Player« ausgeliefert, der — inklusive Ihrem Video — frei kopiert und weitergegeben werden darf. Somit steht einer Verbreitung Ihrer Kunstwerke nichts im Wege.

Mitgeliefert wird ein leider nur englischsprachiges Handbuch, das aber durch seine Fülle und den lockeren Stil gut verständlich ist. Deluxe Video ist trotz seines relativ hohen Preises von 248 Mark ein interessantes Werkzeug für jeden, der präsentieren, lehren oder gestalten muß. Nötig sind allerdings meist noch ein Zeichenprogramm (beispielsweise Deluxe-Paint) und eventuell ein Musikprogramm (Deluxe Music Construction Set oder ähnliches). Diese Kombination ist ideal, da alle Programme mit dem IFF-Standard arbeiten und somit keine Probleme beim Einlesen der Daten entstehen. (Ottmar Röhrig/jk)

Anbieter (Adressen und Preise siehe Marktübersicht):
Atlantis, Softwareland, Medien-Center, Jumbo Soft



Die Bilderflut rollt

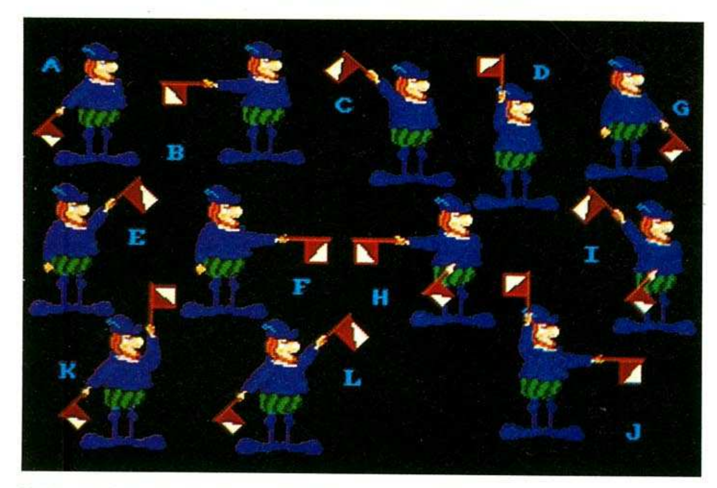


Bild 1. Eine der vielen Grafiken: das Flaggenalphabet

Für den Amiga gibt es seit kurzem zwei neue Disketten mit fertigen Bildern, Brushes und Mustern, die sich speziell an die Malprogrammbenutzer wendet, die wirklich gut gemalte Motive brauchen. Meistens hat man besonders bei der Vorbereitung von Bildern und Animationen für Präsentationen wenig Zeit, diese erst selbst zu erstellen. In diesem Fall bieten sich Grafikbibliotheken wie die Raster Pics von Melzer an. Zum Preis von 69 Mark erhält man zwei Disketten, auf denen in mehreren Schubladen 45 Bilder mit 600 Objekten und 300 Muster der verschiedensten Schattierungen vorhanden sind (Bild 1). (jk)

J.-J. Melzer, Postfach 41 07 27, 1000 Berlin 41, Tel. 030/7 96 71 88

Commodore verstärkt Sportaktivitäten

Dieter Hoeneß, bisher bulliger Mittelstürmer beim Deutschen Meister FC Bayern München, wird im Anschluß an seine aktive sportliche Laufbahn bei Commodore Büromaschinen GmbH neuer Leiter der Sport-Marketing Deutschland. Die neue Aufgabe wird Dieter Hoeneß am 1. Oktober antreten. Er soll sich im wesentlichen um die Sportaktivitäten der deutschen Commodore-Gesellschaft kümmern. Dazu gehöre die Wahrnehmung der Kontakte zu den Commodore-Werbepartnern im Sport, in erster Linie zum FC Bayern München und zur Damen-Volleyball-Mannschaft von Bayern Lohhof. Weiterhin wird Dieter Hoeneß alle übrigen sportlichen Ereignisse und Engagements der Commodore Deutschland GmbH verantwortlich leiten.

In die sportliche Offensive auf europäischer Ebene will Commodore mit dem neuen Verantwortlichen für die internationalen Aktivitäten der Europazentrale, Heinz A. Hocke, gehen. H. Hocke hatte in den letzten zweieinhalb Jahren das Engagement des erfolgreichen Computerherstellers maßgeblich bestimmt. (aa)

SAM-Basic kommt

In letzter Minute erreichte uns die Meldung über die Veröffentlichung des neuen »Simons Amiga Multitasking Basic«. Alte Computerhasen aus der C 64-Szene werden schon erkannt haben, wer der Schöpfer dieser Basic-Version ist: Es ist tatsächlich David Simons, der Programmierer des sehr erfolgreichen Simons Basic für den Commodore 64. Hergestellt wird SAM-Basic in England von PCC Ltd., 3 Mundells Court, Welwyn Garden City, Herts, AL7 1EN. Zur Zeit liegt nur der englische Preis vor: 86,95 Pfund. Sobald Näheres bekannt ist, werden wir darüber berichten. (jk)

Vizawrite-Neuigkeiten

Vor der endgültigen Komplettierung der Textverarbeitung Vizawrite für den Amiga wurde der Preis vom deutschen Vertrieb von 498 auf 198 Mark herabgesetzt. Damit dürfte das Programm auch für Computerbesitzer mit kleinem Geldbeutel interessant werden. Das Programm soll in etwa vier Wochen zur Auslieferung kommen. Der äußerst günstige Preis ist durch eine geänderte Marketingstrategie des Herstellers zu erklären. Zudem will man auch nicht denselben Fehler wie bei Vizawrite 64 machen, das zwar das meistgebrauchte Programm für den Commodore 64 ist, aber nicht das meistverkaufte. Der attraktive Preis soll eine große Käuferschicht ansprechen. Sehr lobenswert ist in diesem Zusammenhang, daß die Käufer, die schon bestellt haben, auch in den Genuß des neuen Preises kommen. An den Möglichkeiten des Programms hat sich ausschließlich in positiver Richtung etwas geändert. Es handelt sich also nicht um eine abgespeckte Version. (jk)

DTM, Bornhofweg 5, 6200 Wiesbaden, Tel. 0 61 21/40 79 89

Mail-Order für den Amiga

In München wurde ein neuer Software-Versand für den Amiga eröffnet. »Jumbo Soft« nennt sich diese Firma, die im wesentlichen nach dem Prinzip eines Mail-Order-Versands funktioniert. Dabei wird besonders auf die schnelle Versendung der georderten Software Wert gelegt. Es gibt bereits einen zirka 40 Seiten starken Software-Katalog, der nicht nur die einzelnen Artikel auflistet, sondern zu jedem Programm eine kurze Beschreibung mitliefert. Außerdem sind bereits 21 Demo-Disketten für die verschiedensten Programme zum Preis von jeweils 12 Mark erhältlich. Es empfiehlt sich die Verwendung von Verrechnungsschecks, dann wird laut Firmenaussage der Bestellprozeß wesentlich beschleunigt. Der Versandkostenanteil beträgt 6,50 Mark, bei Auslandsbestellungen 12 Mark. Es gibt zwar typisch für einen Mail-Order-Versand keinen weiterführenden Service für die gekaufte Software, aber es soll besonderer Wert auf eine schnelle Auslieferung der Programme gelegt werden. (jk)

Jumbo Soft, Software Verlag GmbH, Horemannstraße 2, 8000 München 19, Tel. 089/ 1 23 40 65

Variable Speichererweiterung mit maximal 512 KByte

Eine neue Speichererweiterung für den Amiga ist jetzt in verschiedenen Versionen erhältlich (Bild 5). Die MTR-512-Karte kann wahlweise mit RAMs oder EPROMs bestückt werden. Der Expansionsport ist durchgeschleift und somit können mehrere Karten hintereinandergehängt werden. Sogar das Sidecar läßt sich weiterverwenden. Die Anfangsadresse ist in den 16-MByte-Schritten des Prozessors einstellbar. Die Karte gibt es sowohl als Leerplatine (110 Mark), als Bausatz (178 Mark) oder als Fertigkarte (248 Mark). Voll aufgerüstet mit RAMs auf 512 KByte kostet sie zur Zeit 589 Mark. (jk)

R. Tröps, Pingsdorfer Straße 141, 5040 Brühl, Tel. 0 22 32/1 30 63

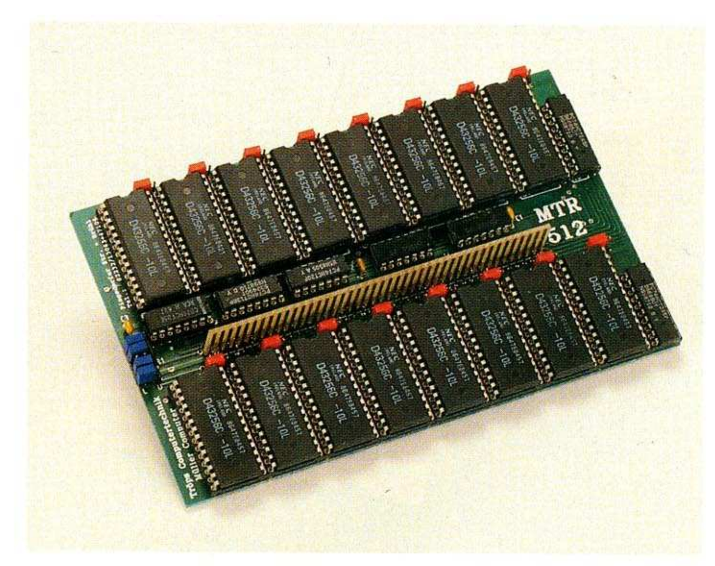


Bild 2. Die Tröps-RAM-Platine voll aufgerüstet

Neue Wege gehen

Wie sieht eigentlich ein Händler die momentane Entwicklung auf dem Soft- und Hardwaremarkt für den Amiga? Was für Tips kann er zukünftigen Profi-Programmierern geben. Wir interviewten Klemens Baumgärtel, einen Mitgründer der Vertriebsfirma Atlantis zum Thema.

Der Markt ist offen, wie eh und je. Ganz besonders gilt dies natürlich im Bereich Amiga Soft- und Hardware. Wer jedoch glaubt, daß man nicht auch hier neue Wege beschreiten kann, der irrt. Die deutsche Vertriebsfirma Atlantis ist noch relativ neu am Markt und trotzdem schon eine der aktivsten, wenn es darum geht, Amiga-Produkte preiswert für den deutschen Anwender bereitzustellen.

Amiga-Magazin: Wie entstand Atlantis und was ist das Neue an Ihrem Konzept für den Vertrieb von Amiga Produkten?

Atlantis: Unsere Firma ist ursprünglich aus einer »Electronik GmbH« entstanden. Durch das Interesse der Mitbegründer am Amiga und durch ein erstes eigenes 3½-Zoll-Laufwerk sind wir richtig auf den Amiga eingestiegen und vertreiben inzwischen nicht nur Hardware, sondern auch Software und Zubehör jeglicher Art. Sehr stolz sind wir zum Beispiel auch über so kleine aber für den Amiga-Benutzer nützliche Sachen wie etwa Diskettenaufkleber, die wirklich genau auf die 3½-Zoll-Disk passen. Es war nicht einfach, in Deutschland einen Hersteller zu finden, der tatsächlich unsere Wünsche umsetzen konnte. Wir versuchen im wesentlichen nicht nur ein einfacher Vertrieb zu sein, sondern beschäftigen uns auch mit den Problemen der Benutzer. Das unterscheidet uns von vielen anderen Firmen, die nur auf Verkauf aus sind. Dazu gehört unter anderem auch, daß wir Public Domain-Programme jeglicher Art nicht nur bereitstellen, sondern sogar eine eigene Reihe ins Leben gerufen haben, auf der man Sachen findet, die sonst auf keiner anderen Diskette zu haben sind. Ganz nebenbei versuchen wir natürlich immer die interessantesten Angebote für den Käufer auf Lager zu haben. Da wir viele Produkte direkt aus den USA beziehen, dürfte uns das nicht schwerfallen.

Amiga-Magazin: Da wir gerade beim Thema Verkauf sind. Welche Produkte für den Amiga laufen am besten?

Atlantis: Sehr gut hat sich bei uns die Einführung von deutschen Handbüchern bezahlt gemacht. Viele Anwender stehen mit einer Original-Anleitung doch recht hilflos da und wir versuchen mit professionellen selbsterstellten und übersetzten Anleitungen auszuhelfen. Ich denke etwa an unser DPaint II-Handbuch. Wegen der hervorragenden Grafikeigenschaften des Amiga hat sich in letzter Zeit auch sehr viel Interesse an Digitalisierern beispielsweise dem preiswerten »Digi View« aufgetan. Überrascht sind wir natürlich auch von dem starken Interesse an guten Utilities von deutschen Programmierern. Der »Boot Boy« liegt dabei offensichtlich genau richtig (Bild). Damit kann man eigene Vorspänne erstellen, die im Boot-Sektor einer Diskette abgelegt werden und die nach dem Einlegen der Disk in das Laufwerk nach spätestens einer Sekunde sichtbar werden.

Amiga-Magazin: Sie arbeiten auch mit deutschen Programmierern direkt zusammen?

Atlantis: Wir kennen überhaupt die gesamte Szene recht gut und versuchen, Talente entsprechend zu fördern.

Amiga-Magazin: Um noch einmal kurz auf die Public Domain-Disketten zurückzukommen. Viele behaupten, die Händler würden sich daran eine goldene Nase verdienen.

Atlantis: Es ist grundsätzlich verboten, Gewinne mit Public Domain zu machen. Aber man muß die Disketten einfach für einen geringen Betrag abgeben. Schließlich sind darin die Kosten für die Disk selbst enthalten und außerdem kann man die anfallende Menge von Bestellungen ohne einen gewissen Zeitaufwand gar nicht bewältigen. Hinzu kommt noch der Materialverschleiß der Laufwerke. Wir planen eventuell jemanden anzustellen, der sich dann nur noch mit der Erstellung und dem Versand dieser Disketten beschäftigt. Alles in allem wird man sich niemals mit solchen Disketten eine goldene Nase verdienen. Uns kommt es darauf an, daß durch die Public Domain-Programme vielen Amiga-Besitzern geholfen werden kann kleine Probleme zu lösen und nebenbei eine Menge zu lernen. Zur Zeit haben wir etwa 170 Disketten im Programm.

Amiga-Magazin: Wie steht es mit dem Erstellen von Handbüchern; auch das kostet doch sicherlich eine Menge Zeit?

Atlantis: Handbücher sind grundsätzlich nicht so sehr kosten- dafür um so zeitintensiver. Für die normalen Handbücher wird der Benutzer beim Kauf des Programmes bei uns allerdings keinen Aufpreis bezahlen. Erst wenn eine Anleitung zum Beispiel auf den Umfang von DPaint II anwächst, verlangen wir einen entsprechenden Anteil. Das nächste größere Handbuch, was wir uns vornehmen, wird Deluxe Video 1.2 sein.

Amiga-Magazin: Sehen Sie bei Atlantis aufgrund der verschiedenen Amiga-Modelle Probleme bezüglich der Kompatibilität der bereits erhältlichen und noch kommenden Software?



Nach wenigen Sekunden sichtbar: ein Boot-Boy-Vorspann

Atlantis: Es bleibt abzuwarten, wie gut sich die Hersteller auf die verschiedenen Amiga-Modelle einstellen werden. Für uns als Händler gilt die Regel, daß wir zuerst jegliche Software genauestens testen, und zwar auf allen drei Modellen. Erst dann können wir Anfragen wegen einer eventuellen Nicht-Kompatibilität beantworten.

Amiga-Magazin: Zum Schluß noch eine Frage, die sicherlich alle Noch-Hobbyprogrammierer interessiert, die aber vielleicht einmal ihre Programme professionell verkaufen möchten. Wenn diesbezüglich jemand an Atlantis herantritt. Was kann er erwarten?

Atlantis: Grundsätzlich müssen wir vorab ganz klar sagen, und das mag für einige im ersten Moment vielleicht etwas hart klingen, zahlen wir keinen Vorschuß auf kommende Produkte. Dies entspringt unserer Philosophie, die gesamte Szene aus Programmierern oder sogar ehemaligen Crackern und Hackern zwar zu einer vernünftigen Zusammenarbeit zu bewegen, jedoch nicht zu erpressen. Deswegen haben wir ein sogenanntes Amiga-Forum gebildet, in dem interessierte Leute zu einer fruchtbaren Zusammenarbeit zusammenfinden können. Von uns bekommen die Programmierer aber die entsprechende Unterstützung, um überhaupt vernünftig arbeiten zu können. Dies bezieht sich auf das Bereitstellen von Programmiersprachen wie zum Beispiel teure C-Compiler und Assembler. Außerdem können wir natürlich auch in bestimmten Rahmen Hardware wie Speichererweiterungen, Hard-Disks oder auch Digitizer zur Verfügung stellen. Da wir sehr viele Leute kennen, können wir optimal zwischen verschiedenen Talenten vermitteln. Wenn zum Beispiel jemand sehr gute Grafiken erstellt und wir außerdem noch einen Musikprofi an der Hand haben, dann werden die entsprechenden Kontakte zu einem Spieleprogrammierer hergestellt, der sich nicht so gut mit solchen Spezialgebieten auskennt. Am Ende kommt hoffentlich etwas Gutes dabei heraus.

Amiga-Magazin: Wir danken Klemens Baumgärtel, der dieses Gespräch stellvertretend für Atlantis mit uns geführt hat.

(jk)

dbMAN - ein Datenriese der alten Schule

Dank des großen Befehlssatzes und der Kompatibilität zu dBase III ist die Datenbanksprache dbMAN inzwischen recht beliebt geworden. Während sich das Programm auf dem IBM-PC und Atari ST bereits bewährt hat, zeigt die Amiga-Version noch einige Schwächen in verschiedenen Bereichen.

Wie auch dBase gehört dbMAN zu den Datenbanksprachen; ist also eine Programmiersprache speziell auf die Probleme der Dateiverwaltung zugeschnitten. Durch die außergewöhnliche Anzahl von Befehlen und Funktionen hat der Programmierer ein mächtiges Werkzeug an der Hand. Mit einer Kapazität von bis zu 2 Billionen Datensätzen pro Datei, wobei sich ein Datensatz aus maximal 128 Datenfeldern mit insgesamt 4000 Zeichen zusammensetzen kann, läßt sich schon einiges anfangen. Für einen Preis zwischen 319 und 548 Mark erhält der Anwender eine Programmdiskette mit einem umfangreichen englischen Handbuch.

Dieser Anwender sollte sich auf dem Gebiet der Datenbanksprachen auskennen. Für einen Neuling auf diesem Gebiet ist das Handbuch denkbar ungeeignet; die einzige Anweisung, die er beim Durcharbeiten der ersten Seiten unmittelbar nachvollziehen kann, ist die Aufforderung, das Programm durch die Eingabe von »dbMAN« zu starten. Das Handbuch bietet dem Profi in mehreren Kapiteln geballte Information zu bestimmten Schwerpunkten und ist damit eher als Nachschlagewerk zu bezeichnen.



Bild 1. dbMAN — eine Datenbanksprache mit umfassender Dokumentation

Beim Umgang mit dbMAN auftretende Mängel trüben jedoch die Begeisterung für dieses sonst so leistungsfähige Programm. So stellt man bereits nach ersten Gehversuchen fest, daß die Originaldiskette nicht automatisch startet (bootet). Nach einer näheren Untersuchung der Diskette mit dem CLI fanden wir zwei Textdateien, von denen die erste beschreibt, wie Daten aus dem Bestand der Datei selektiert und für eine Ausgabe vorbereitet werden. Die zweite Datei weist auf Einschränkungen beim Ablauf von dbMAN mit einem 512K-Amiga hin. Das zeigt, daß dbMAN auf dem Amiga nicht optimal programmiert ist, denn andere Computer nehmen das Programm auch mit so »wenig« Speicher vollständig auf. Nach einigen Bemühungen gelang es, eine bootfähige Sicherheitskopie zu erstellen. Probleme treten hier auf, weil nirgends beschrieben wird, welche Libraries der Workbench von dbMAN verwendet werden.

Als sehr unvorteilhaft stellt sich die Bedienerführung auf dem Grafikwunder Amiga dar. Auf die Verwendung der Maus und anderen Komfort, den die Benutzeroberfläche Intuition mit Menüs, Schaltsymbolen und Entscheidungsfenstern bieten kann, wurde bei dbMAN gänzlich verzichtet. Alle Eingaben erfolgen, ohne die Möglichkeit der Abkürzung von Befehlen, über die Tastur meist als Kontroll-Steuersequenzen (die Taste <CTRL> und ein weiteres Zeichen); seltener mit den Funktionstasten. Auch die Ausgabegeschwindigkeit der Texte kann noch erheblich verbessert werden.

Programmierer der alten Schule können durch dbMAN ihre Kenntnisse in der Datenbanksprache nun weiter auf dem Amiga einsetzen. Sie sind mit Kontrollsequenzen aufgewachsen und können sich eine andere Arbeitsweise vielleicht nicht vorstellen. Programme wie dbMAN haben Softwaregeschichte geschrieben. Die zügige Weiterentwicklung der Hardware hat jedoch eine neue Qualität der Bedienerführung ermöglicht. Nicht zuletzt der Amiga hat gezeigt, daß auch leistungsfähige Programme durch eine gelungene optische Gestaltung einfach zu bedienen sind. Das Lernen unhandlicher Zeichenfolgen für die Programmsteuerung sollte endgültig der Vergangenheit angehören. (Heinz Wrobel/pa)

Anbieter (Adressen und Preise siehe Marktübersicht): Atlantis, B. Soltau Electronics, Instant Replay, Interplan, ITC, Philgerma, Softwareland, Soyka Datentechnik, PDC, Jumbo Soft

Keine Karten

Der für diese Ausgabe geplante große Test der Steckkarten für den Amiga 2000 mußte leider ausfallen. Bisher war nur die XT-Karte in der tatsächlichen Verkaufsversion verfügbar. Sobald mehrere verschiedene Karten auf den Markt kommen, werden wir dieses Thema mit Sicherheit ausführlich aufgreifen. (jk)

Plattenirrtum

Im Amiga-Magazin 6/7 unterlief uns auf Seite 109 ein Irrtum bezüglich der Bezugsadressen für die C Ltd.-Festplatte. Den Exklusiv-Vertrieb für die Festplatten von C Ltd. hat in Deutschland Intelligent Memory, Basaltstraße 58, 6000 Frankfurt 90, Tel. 069/708525. (jk)

Amiga am Farbfernseher

Von Commodore wird zum Anschluß eines Farbfernsehers an den Amiga das Modul A520 MOD angeboten (Bild 2). Es wird an den RGB-Ausgang und die Audio-Ausgänge des Amiga angeschlossen. Ein normales Koaxialkabel dient zur Verbindung mit dem Antenneneingang des Fernsehers. Die Qualität des Bildes ist bei hochauflösender Grafik und auch bei 80-Zeichendarstellung erstaunlich gut. Man kann also auch längere Zeit auf dem Farbfernseher arbeiten, ohne Kopfschmerzen zu bekommen. Durch den Anschluß der zwei Audio-Stecker an das Modul wird der Ton mono an den Fernseher übertragen (nur mono).

Der zweite Ausgang des Moduls dient zum Anschluß an einen Videoeingang, zum Beispiel eines Videorecorders. Der Preis von 58 Mark ist für die Einsparung des Commodore 1081-Monitors, der ja fast 1000 Mark kostet, auf alle Fälle gerechtfertigt. (rb)

Bezugsquelle: autorisierter Commodore-Fachhandel



Bild 2. Endlich erhältlich: Das Amiga Fernseh-Modul

Digi-View Update

Für alle, die den Digitizer »Digi-View« von New Tek besitzen, gibt es jetzt die neue Software-Version 2.0. Auffallendste Neuerungen sind die verbesserten Menüs und Digitalisierungsmöglichkeiten in mehreren Auflösungen; unter anderm auch ein H.A.M.-Modus mit verbesserter Berechnung (4096+). Auch Einstellungen verschiedener Kameras sind kein Problem mehr. Diese Version wird mit dem Set ausgeliefert. Außerdem kann man bei Atlantis, Ernst-Reuter-Straße 151, 5060 Hürth 4, Tel.: 02233/31066, die neue Software gegen einen Aufpreis von 38 Mark beziehen. Dieser Preis gilt allerdings nur für Besitzer der alten Version 1.0, die ihre Diskette einsenden müssen und dann das neue Programm auf Diskette erhalten. (jk)

Beim Verein für Fortbildung in Hannover wird der Amiga als modernes Unterrichtsmittel eingesetzt. Er wird verwendet zur Fortbildung von Grafikern, die nach ihrer Ausbildung gute Chancen haben, einen interessanten Arbeitsplatz in der Industrie zu bekommen.

Amiga im Unterricht

Im Unterrichtsraum stehen zwölf Amiga 1000, die untereinander vernetzt sind. Von diesen Computern arbeitet einer als sogenannter »Master«, die anderen elf sind als »Slaves« geschaltet. Der Lehrer, der am Master sitzt, hat mit einem eigenen Schaltpult viele Möglichkeiten, die Ausgaben der Computer umzuleiten. Zum Beispiel kann er den Bildschirminhalt eines beliebigen Computers mit einem Großbildprojektor anzeigen und Erläuterungen dazu geben. Er ist jedoch genauso in der Lage, das Bild auf allen Slaves darzustellen, wobei die Arbeit des einzelnen nur kurzzeitig unterbrochen und nicht durch Datenverlust zerstört wird.

Auch das Ansehen des Bildschirms einzelner Slaves auf dem Master ist kein Problem und ermöglicht die genaue Kontrolle der Arbeiten. Der Lehrer kann sogar einzelne Computer so abschalten, daß auf ihnen überhaupt nicht mehr gearbeitet werden kann. Einspielungen von Bildern, die mit der Videokamera aufgenommen werden, lassen sich sowohl mit dem Projektor als auch auf den Slaves verwirklichen. Durch ein Mikrofon kann der Lehrer alle oder einen einzelnen Schüler ansprechen, wobei die Lautsprecher der Monitore verwendet werden. Die Lautstärke kann der Schüler individuell regeln. Die zwei zur Verfügung stehenden Farbdrucker können von allen Computern aus ohne Schwierigkeiten benutzt werden.

Die Schüler (Bild 3) haben sich sehr schnell an den neuen Unterricht und den Computer mit dem Grafikprogramm gewöhnt. Inzwischen sind sie in der Lage, Bilder in wenigen Stunden zu erstellen, für die sie früher mit »normalen« Werkzeugen Tage und mehr benötigten. Interessant ist auch die technische Seite der Anlage, die mit sogenannten »Clusterboxen« arbeitet. Jeder der Slaves hat eine solche Box, die vom Schaltpult bei dem Master gesteuert werden kann. Die Anschlüsse für den Monitor, den Drucker und die Lautsprecher laufen durch diese Box. Über sehr gute Relais können nun die einzelnen Signale umgeleitet werden. Dabei mußten natürlich verschiedene technische Probleme gelöst werden, zum Beispiel ist die Übertragung des RGB-Signals über ein Koaxialkabel ziemlich problematisch. Aber seit der Installation läuft das System problemlos und wird von den Schülern voll akzeptiert.

Alles in allem hat sich das neue System gut bewährt und wird weiter verfolgt. (aa)



Foto: Commodore

Bild 3. Schnelleres und leichteres Arbeiten durch Einsatz von Computern als modernes Unterrichtsmittel

Fachwortschatz

1) Ich lese immer wieder die Begriffe »Fast RAM« und »Chip RAM«. Was ist eigentlich der genaue Unterschied? Mir wird öfter gesagt, daß ich eine größere Anzahl von Grafikdaten als 512 KByte erst mit dem Blitter in das Chip-RAM kopieren muß. Stimmt das?
2) Ich komme trotz Ihres Artikels zur Workbench nicht mit den verschiedenen Fachbegriffen zurecht. In jeder Software-Anleitung steht etwas anderes. Was bedeutet zum Beispiel »Anforderungsfenster«?

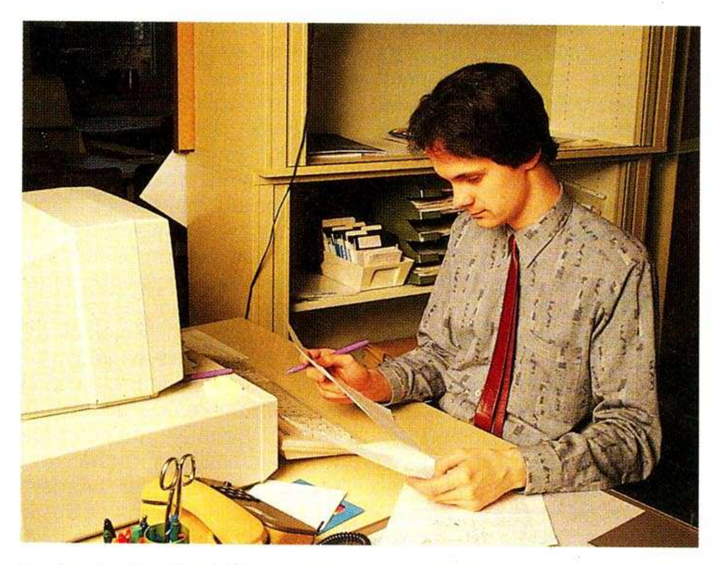
ANDREAS OBERMÜLLER

1) Der Speicher des Amiga wird grundsätzlich in die Arten des Speicherzugriffs eingeteilt; das Chip-RAM ist der Bereich des Speichers, den die Custom-Chips benutzen können. Das Chip-RAM kann auf den gegenwärtigen Modellen der Amiga-Serie nicht höher als 512 KByte sein (in Amerika bastelt man aber schon an neuen Chips, die 2 MByte Chip-RAM haben). Das heißt, man kann immer nur höchstens 512 KByte Daten für Grafik und Sound gleichzeitig verwenden, auch wenn eine 2-MByte-Erweiterung angeschlossen ist. Wenn in diesem größeren Speicher mehr Grafikdaten sind, müssen sie immer erst in diesen 512-KByte-Grafikspeicher geschaufelt werden, um sie anzuzeigen. Wer sich nun aber denkt, das schnelle Datenschaufeln würde mit dem Blitter vonstatten gehen, der ist vielleicht etwas falsch informiert. Der Blitter ist auch nur Teil eines Custom-Chips und kann deshalb nur Daten innerhalb der unteren 512 KByte kopieren. Grafikdaten, die nicht im Chip-RAM, sondern im Fast-RAM liegen, können also nur vom Hauptprozessor kopiert werden. Das Fast-RAM ist also nur für den 68000-Prozessor zugänglich, nicht aber für die Custom-Chips. Dafür aber ist es schneller als das Chip-RAM, deswegen der Name Fast-RAM. Das Chip-RAM muß abwechselnd den Zugriff der Custom-Chips und des 68000 erlauben, während im Fast-RAM der Prozessor nicht von den Custom-Chips gebremst werden kann.
2) Es ist schon ein Jammer, daß sich die Softwarehersteller und Buchautoren nicht auf einheitliche Übersetzungsrichtlinien einigen können. Commodore gibt zwar solche Listen an Entwickler aus, um ein Tohuwabohu zu vermeiden, doch die Übersetzungen aus diesen Richtlinienpapieren sind auch nicht immer gerade die besten. Das Wort, das Sie nennen, ist ein sehr gutes Beispiel dafür. Im englischen Original heißt das Gesuchte »Requester box«. Die Beschreibung des Requesters finden Sie übrigens in der letzten Spalte des Workbench-Artikels auf Seite 26 der Ausgabe 6/7 des Amiga-Magazins. Wir werden in einer der nächsten Ausgaben eine Liste von uns bekannten Übersetzungen veröffentlichen. Für Anregungen im Bereich (guter) Übersetzungen haben wir im übrigen immer ein offenes Ohr.

(M. KOHLEN/jk)

FRAGEN SIE

Wenn Sie Probleme mit dem Amiga oder mit anderen Computern in Zusammenarbeit mit dem Amiga haben, stellen Sie Ihre Fragen ans Leserforum des Amiga-Magazins. So können Sie in Kontakt mit anderen Lesern kommen, die vielleicht dieselben Probleme schon gelöst haben.



In der Amiga-Redaktion bearbeitet unser Redakteur Jörg Kähler Leserpost für das neue Leserforum

Der Trick mit den Icons

Wenn man die Icons auf der Workbench anklickt, verändern sie ihre Farbe. Ich habe inzwischen auf mehreren Utility-Disketten Icons entdeckt, die sich beim Anklicken bewegen oder ein völlig anderes Aussehen annehmen. Wie kann man solche Icons selber konstruieren?

KLAUS MAREI
Ausgabe 6/7

Grundsätzlich kann man Icons mit dem Icon-Editor, der sich auf jeder Workbench-Diskette in der Schublade »System« befinden sollte, entwerfen. Eine Bedienungsanweisung finden Sie im Amiga-Handbuch (Markus Breuer, Markt&Technik Verlag) auf Seite 357. Im Normalfall ändern sich bei Anklicken eines solchen einfachen Icons die Farben innerhalb des Icons folgendermaßen: Die erste Farbe aus dem Preferences-Menü wird zur vierten, die zweite zur dritten, die dritte zur zweiten und die letzte zur ersten Farbe. Nun gibt es aber auch die Möglichkeit, ein sogenanntes »Double Image-Icon« oder, auf deutsch, Doppel-Icon zu produzieren. Die »bewegten« Icons sind dabei dasselbe wie die mit völlig anderem Aussehen, denn bei Anklicken eines solchen Icons wird einfach auf ein anderes Icon umgeschaltet. Um ein Doppel-Icon zu erzeugen, muß man zwei Icons mit dem Icon-Editor malen; die so abgespeicherten zwei Icons werden mit einem kleinen Utility zu einem »info«-File verarbeitet, das jetzt als Doppel-Icon funktioniert. Auf der überall erhältlichen Public-Domain-Diskette »Fish Disk 12« befindet sich ein solches Utility (SetAlternate) samt Source-Code in Lattice-C.

OLIVER GRÜNDEL

Probleme in C

Aus einigen Beispielen von Public-Domain-Disketten habe ich mir Wissen über die Funktionsweise der Systemprogrammierung erworben. Inzwischen scheitere ich aber an dem Problem der verwendeten Fachbegriffe. Die Entwicklerdokumentation ist mir zu teuer. Deshalb die folgenden Fragen an Sie: Auf was zeigt der Zeiger »ImageShadow« und mit welchen Programmkniffen kann ich festlegen, welche Objekte miteinander kollidieren dürfen und welche nicht?

PETER PUDORFER

ImageShadow zeigt auf die sogenannte ShadowMask eines Bobs. Die ShadowMask ist eine logische Oder-Verknüpfung aller gesetzten Bits eines BOBs, also praktisch die Schwarzweiß-Form des BOBs (Nullen nur dort, wo Hintergrundfarbe ist). Das Betriebssystem verwendet diese Maske im Zusammenhang mit dem »PlaneOnOff«-Parameter zum Farbensetzen. PlaneOnOff sagt dem System, was mit den Bitplanes des Bobs zu tun ist, die nicht durch »PlanePick« angewählt wurden. Wenn Sie die Beispiele, die Sie haben, ausreichend begutachtet haben, müßten Sie bei einem Vorkommen von ImageShadow allerdings auch die anderen Werte entdeckt haben. Kollisionen können Sie durch Setzen einer sogenannten »CollisionMask« kontrollieren, dazu wird ein Zeiger »CollMask« auf diese Maske gesetzt. Sie sehen schon, um ausführliche Dokumentation (ROM Kernel Manual) kommen Sie nicht herum. Schwierigkeiten kann es dabei dann nur noch geben, wenn Sie nicht sehr gut Englisch können. MANFRED KOHLEN

Sprites und Compiler

Ich versuche seit geraumer Zeit, mit Manx Aztec C und der Hilfe des ROM Kernel Manuals VSprites zu programmieren. Mit den Simple Sprites habe ich keinerlei Probleme mehr, aber irgend etwas scheint im ROM Kernel Manual in bezug auf VSprites falsch zu sein. Da ich noch nicht so lange C programmiere, habe ich nicht die Erfahrung, den Fehler selbst zu finden. Wer hilft mir?

ARMIN NUSER

Textverarbeitung

Ich bin seit 5 Monaten stolzer Besitzer eines Amiga 1000 und noch immer auf der Suche nach einer Textverarbeitung, die deutschen Zeichensatz verarbeitet, preisgünstig ist, und möglichst das Einbinden von Grafiken in den Text erlauben sollte. Man sieht zwar immer wieder Werbung und sogar Softwaretests, doch in den Computershops meiner Umgebung kann ich nichts finden.
GABOR SOLIEV

Inzwischen sind mehrere Textverarbeitungssysteme für den Amiga erhältlich; einige sind in Vorbereitung. Die Programme »Scribble!« und »TextCraft« sind im Versandhandel erhältlich — erkennen aber in den gegenwärtigen Versionen entweder den deutschen Zeichensatz nicht oder umgehen das Problem der amerikanischen Programmierer mit seltsamen Tastenkombinationen. Die erste vollwertige deutsche Textverarbeitung ist »UBM-Text« von UBM-Drecker GmbH; sie verarbeitet allerdings keine Bilder. Zwei hervorragende Programme sind aber derzeit im Anmarsch — Vizawrite Amiga und ProWrite. Einen Kurztest der beiden Programme finden Sie in Amiga 6/7-87. Beide Programme sind voraussichtlich noch in diesem Herbst erhältlich. (jk)

Maschinensprache

Ich möchte von 68000-Maschinensprachecode aus die AmigaDOS-Routinen aufrufen, und bin dabei auf zwei Möglichkeiten gestoßen, von denen allerdings nur eine stimmen kann. Entweder das AmigaDOS-Developers-Manual (jetzt Teil des AmigaDOS Manual, Bantam Books, die Red.) hat unrecht oder der Programmierer, dessen Programm mir als Vorlage diente. Das Manual sagt, LVO_ sollte als Prefix für die EXEC-Funktionen verwendet werden, im Beispielprogramm steht statt dessen _LVO. Was stimmt?
MICHAEL ASBLER

Tatsächlich besteht dieser Fehler im AmigaDOS-Manual; der Programmierer hat recht (würde sonst sein Programm funktionieren?). Der Underline-Strich muß VOR dem LVO stehen, an das LVO anschließend (ohne Leerraum!) der Name der EXEC-Routine. Übrigens sind die ROM EXEC-Routinen keine AmigaDOS-Routinen — da haben Sie sich wohl nur verschrieben?
(ANDREAS EPFEL/rb)

Der zweite Klick

Warum kann man ein zweites CLI-Fenster nur mit dem Befehl NEWCLI aufrufen? Es nutzt nichts, einfach das CLI-Icon in der Systemschublade nochmals anzuklicken, wenn man dies schon einmal getan hat.
RAINER TASCHKE
Ausgabe 6/7

Offenbar hat Herr Taschke entweder eine steinalte Workbench-Version oder schlechte Augen. Probieren Sie es doch mal aus: Nochmaliges Klicken eröffnet einen neuen CLI. Wenn man genau hinsieht, merkt man sogar, daß sich ein neues Window öffnet; wenn Herr Taschke das Window des laufenden CLI nicht geändert hat, erscheint das neue CLI-Window natürlich an genau derselben Stelle in derselben Größe. Daß man in einem neuen CLI ist, bemerkt man übrigens an der geänderten Zahl im Prompt (Nummer des Tasks).
(ANDREAS EPFEL/jk)

Fragen zum Amiga 2000

1) Ich habe gehört, der Amiga 2000 wird mit einem schlechteren Monitor geliefert als der Amiga 1000. Ist das wahr?
2) Ich will mir den Amiga 2000 kaufen und eventuell später eine PC-Karte dazu. Nun wird mit dieser PC-Karte aber ein 5¼-Zoll-Laufwerk geliefert. Ich habe aber schon ein Industrielaufwerk, das ich mir gerne selbst anschließen möchte. Muß man das Laufwerk mitkaufen oder ist die Steckkarte auch einzeln verfügbar? Wenn sie einzeln verfügbar ist, ist beim Anschluß des Industrielaufwerks etwas Besonderes zu beachten?
PETER SCHMITT

1) Von mehreren Seiten hat man in der letzten Zeit gehört, daß Commodore andere Monitore liefert; ein Gang in verschiedene Computerläden bestätigte in einigen, aber nicht allen Fällen, dieses Gerücht. Offenbar stellt man sich gerade um. Dieses Problem ist allerdings nicht nur auf den Amiga 2000 begrenzt. Der Monitor hat von Commodore aus dieselbe Modellnummer (1081), es handelt sich aber offenbar um ein neues (billigeres) Modell von Philips; auch die bisherigen 1081-Monitore waren Philips-Fabrikate, hatten aber eine qualitativ höherwertige Lochmaske. Offenbar handelt es sich hier um Kostensenkungsmaßnahmen seitens Commodore. M. KOHLEN

2) Soweit uns bekannt ist, kann man die PC-Karte nur zusammen mit Laufwerk ergattern. Inwieweit Sie ein eigenes Laufwerk anschließen können, ist eher ein technisches Problem, das sich nicht mit ein paar Worten lösen läßt. Vielleicht kann ein technisch versierter Leser das Problem bereits lösen? (jk)

Interface benötigt?

Braucht man ein Wiesemann 92000/G Interface, um einen Citizen-Drucker an den Amiga anzuschließen?
DARREN SHAW

Am Amiga lassen sich alle Drucker mit Centronics-Schnittstelle (im Prinzip also alle Epson-kompatiblen oder IBM-Drucker) problemlos anschließen. Dazu wird kein Interface benötigt, da der Amiga auch über eine Centronics-Schnittstelle verfügt. Um einen speziellen Drucker inklusive seiner besonderen Steuercodes voll ausnutzen zu können, braucht man dann nur den passenden Druckertreiber mittels der »Preferences« auf den Disketten softwareseitig zu installieren. Viele Druckertreiber werden auf der Workbench zum Amiga bereits mitgeliefert. Ein fertiges Programm zur Erstellung eigener Druckertreiber gibt es noch nicht. (jk)

Konstruktive Kritik

Mit großem Interesse habe ich die Erstausgabe des Amiga-Magazins gelesen. Auch ich habe ähnliche Erfahrungen gemacht, wie sie das Ehepaar Wolf im Leserforum beschreibt. Lobenswert ist, daß Sie auch Leserbriefe veröffentlichen, die sich über mangelnde Dokumentation auslassen; viele auf einen Computertyp spezialisierten Zeitschriften loben meistens nur den Hersteller in den Himmel.

Aber wenn Sie sich der Probleme dieser Leserschicht — den Anfängern — nicht nur in Form von oft zu ausführlichen Einführungen widmen würden, wäre ich noch viel glücklicher mit dem Amiga-Magazin. Manch einer kann mit Hilfe von übersichtlichen Tabellen schneller lernen, als wenn er sich durch langen Text lesen muß. Natürlich müssen auch solche Artikel geschrieben werden, denn viele Leser brauchen ausführliche Anleitungen. Andere wiederum brauchen einfach das technisch-wissenschaftliche, für unsereinen unverständlicher, Brimborium. Vergessen Sie aber bei aller Sorge um den Ausgleich zwischen Profi- und Anfängerliteratur nicht, daß es zwischen diesen Extremschichten auch noch Leser gibt, die gerade eben das Anfängertum des absoluten Nicht-Verstehens hinter sich gebracht haben, oder derer, die schon fast Profis wären, würde die Information nicht nur für die Absolut-Profis geschrieben.

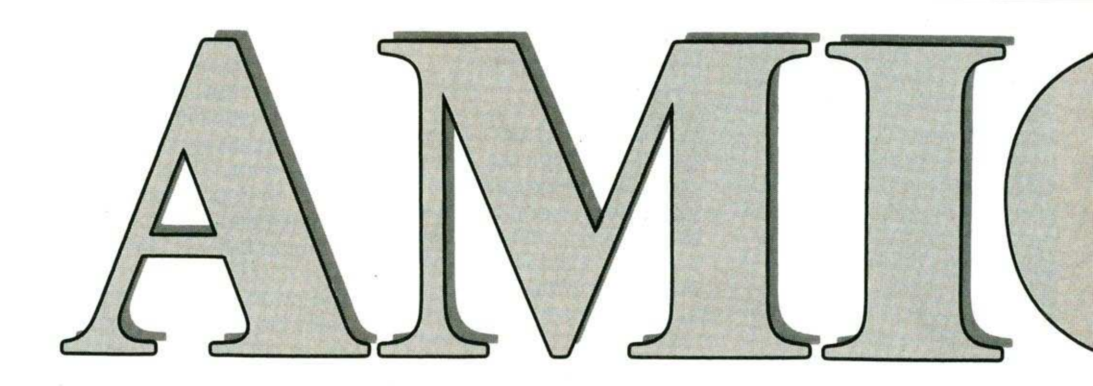
Da kommt mir der Vorschlag des computerbegeisterten Ehepaars gerade recht, um Sie um eine Veröffentlichung von übersichtlichen Tabellen zum Beispiel aller Fehlermeldungen zu bitten — etwas für mich, den »Nicht-mehr-ganz-Anfänger«. Was natürlich keineswegs ausschließt, daß Sie auch mal Betriebssystembefehle in irgendwelchen Listen veröffentlichen — vielleicht verstehe ich es ja später mal, und irgend jemand versteht es jetzt bestimmt, so wie ich die Fehlermeldungen.

Ich hoffe, Sie nehmen sich diese Anregungen zu Herzen.
OTMAR GEISSLER

Das werden wir auf jeden Fall machen.
Die Redaktion

ANTWORTEN SIE

Haben Sie schon eine Lösung zu bestimmten allgemeinen Problemen gefunden, die viele Leser interessiert, dann schreiben Sie uns Ihre Antwort oder Lösungsvorschläge. Durch ein Plus an Informationen kann die Gemeinde der Amiga-Freaks schneller wachsen.


AMIG

PRÄSENTIERT:



DATAMAT Amiga

Der Datenspezialist

DATAMAT Amiga ist die konsequente Umsetzung des erfolgreichen DATAMAT ST. Dabei werden natürlich alle speziellen Amiga-Eigenschaften, wie z. B. Multitasking, unterstützt. So lassen sich jetzt alle Daten auch mit dem Amiga professionell pflegen und verwalten – in der Geschwindigkeit, die man von diesem Rechner gewohnt ist, und mit einem Komfort, den man bei einer Programmsteuerung über Maus und Tastatur erwarten darf.
DATAMAT Amiga in Stichworten:
Maximal 8 offene Dateien gleichzeitig – Dateigröße: max. 2 Milliarden Zeichen – Datensatzgröße: max. 64.000 Zeichen – maximal 2 Milliarden Datensätze – Anzahl der Datenfelder: unbegrenzt – maximale Feldgröße: 32.000 Zeichen – Paßwort-Schutz – Dateien werden auf Massenspeicher (hohe Datensicherheit) bearbeitet – maximal 80 Indexfelder mit wählbarer Genauigkeit (1–999 Zeichen) – komfortable Such- und Selektierkriterien (Bereiche, Und-, Oderverknüpfung ...) – Text-, Datums-, Zeit-, Zahlen- und Auswahl-Felder, Einlesen von IFF-Dateien – Datenaustausch mit anderen Programmen möglich, wichtig für Serienbrieferstellung – Programmsteuerung über Maus und Tastatur – frei gestaltbare Bildschirmmaske, bis zu maximal 5000 x 5000 Punkte groß – Bildschirmmaske unterstützt Grafikelemente wie Rechteck, Kreis, Linie, Muster usw. – Bildschirmmaske unterstützt verschiedene Textarten und -größen – mehrzeilige Textfelder mit Wortumbruch und Formatierungsmöglichkeiten – integrierter Druckermasken- und Listeneditor.

DATAMAT Amiga DM 99,–

PROFIMAT Amiga

Das komplette Entwicklungspaket

Viele wissen um die fantastischen Möglichkeiten eines echten Assembler-Programms, wagen sich aber an eigene Programme selbst nicht heran. Dabei ist die Maschinensprache-Programmierung durchaus nicht nur ambitionierten 68000-Experten vorbehalten. Schnell erlernbar und überaus bedienerfreundlich steht jetzt jedem Anwender ein komplettes Entwicklungspaket zur Verfügung, mit dem er seinen Amiga voll ausreizen kann: PROFIMAT Amiga.
PROFIMAT Amiga in Stichworten:
Komplett in Assembler geschrieben, daher extrem schnell – integrierter Editor, Debugger, Disassembler und Reassembler – umfangreiche Betriebssystem-Bibliothek – läuft unter CLI und Workbench – erzeugt optional PC-relativen oder absoluten Code – Makros mit nahezu beliebig vielen Parametern unterschiedlicher Typen möglich – Fehlersuchfunktion – Cross-Referenz-Liste – bedingte und wiederholte Assemblierung menügesteuert – volle 32-Bit-Arithmetik – Debugger mit 68020-Single-Step-Emulation – lauffähig auf jedem Amiga mit 512 KByte und Kickstart 1.2.

PROFIMAT Amiga DM 99,–
erscheint ca. 8/87

geringfügige Änderungen vorbehalten

BESTELL-COUPON

Einsenden an: DATA BECKER · Merowingerstr. 30 · 4000 Düsseldorf 1
Bitte senden Sie mir:

zzgl. DM 5,– Versandkosten
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name

Straße

Ort



Welcher Amiga

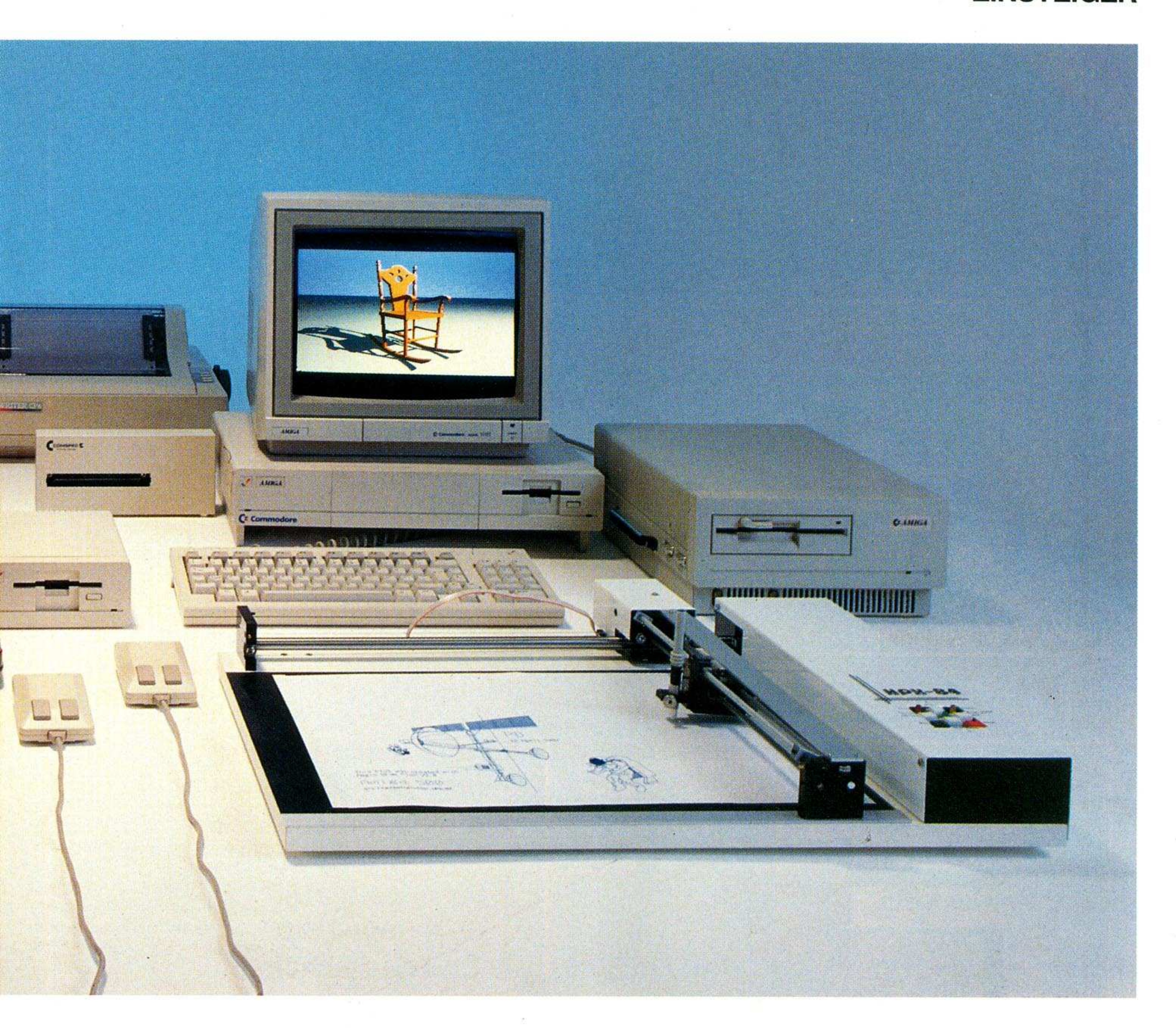
Die Amiga-Palette umfaßt derzeit drei verschiedene Systeme: Amiga 2000. Wir sagen Ihnen, welcher für Sie der richtige ist. Gerät aus der Amiga-Palette Sie sich entscheiden können, um

Mittlerweile hat ein Käufer, der sich für den Amiga interessiert, die Auswahl zwischen drei verschiedenen Computern: Amiga 500, Amiga 1000 und Amiga 2000. Doch wodurch unterscheiden sich diese drei Computer? Für welchen Amiga soll sich der Kunde entscheiden, welcher erfüllt die Ansprüche, die an ihn gestellt werden? Wir gehen dieser Frage auf den Grund und sagen Ihnen, welches Gerät für Ihre Anwendungen das geeignetste ist.

Ursprünglich produzierte Commodore zuerst den Amiga 1000, von dem inzwischen über 20000 Stück in deutschen Haushalten stehen. Commodore prophezeite zum Zeitpunkt der Einführung, daß dies nur der Anfang einer Produktpalette sein sollte. Bis heute hat sich der Umfang dieser Serie auf drei verschiedene Gerätetypen erweitert. Dabei geht Commodore davon aus, daß für den Privat- und Geschäftsbereich verschiedene Amiga-Modelle zum Einsatz kommen. So ist der Amiga 500 hauptsächlich als Heimcomputer gedacht. Das macht auch der (Listen-)Preis von derzeit 1295 Mark für den Computer mit eingebauter Floppystation deutlich. Der Amiga 2000 ist dagegen für den Geschäftsbereich konzipiert worden, was man auch klar am Preis von 2995 Mark in der absoluten Minimalausstattung sehen kann.

Der Amiga 1000 ist momentan so etwas wie ein Zwitterwesen. Er läßt sich zwischen den beiden anderen Modellen einordnen, wobei die Tendenz eher in den Heimcomputerbereich zeigt. Der folgende Artikel stellt jedes der drei Modelle mit seinen Stärken und Schwächen vor und gibt Ihnen eine Entscheidungshilfe, welches System Sie denn nun kaufen sollen. Beginnen wir zuerst mit den Gemeinsamkeiten der drei Amiga-Modelle:

Jedes Modell basiert auf dem 16/32-Bit-Mikroprozessor



ist der richtige?

den kleinen« Amiga 500, den Amiga 1000 und das Profigerät Nach Studium dieses Berichts werden Sie wissen, für welches ein Ihren Anforderungen gerechtes System zu erhalten.

68000 von Motorola und wird mit einer Frequenz von 7,14 MHz getaktet. Dieser Prozessor ist derzeit einer der schnellsten, die bei Computern im Heimcomputer- und PC-Markt eingebaut werden. So ist er beispielsweise noch in den Atari-ST-Computern und im Apple Macintosh enthalten.

Außerdem verfügt jeder Amiga über drei zusätzliche Spezial-Prozessoren, die den 68000-Prozessor bei einigen Aufgaben die Arbeit abnehmen. Dies hat den Vorteil, daß die CPU (der Hauptprozessor) nicht unnötig von Aufgaben gebremst wird, die sowieso besser von den Spezialchips erledigt werden.

Der erste der Zusatzprozessoren, »Paula«, ist neben der Kontrolle der RS232-Schittstelle und der Joystick-Ports für den Sound des Amiga verantwortlich. Durch die Eigenschaft, die Hüllkurve selbst zu bestimmen, kann der Amiga alle erdenklichen Töne, Geräusche und Sounds von der Gitarre bis hin zum Vogelgezwitscher perfekt nachahmen. Allerdings ist das Erfassen eines solchen Natursounds nur mit einem speziellen Interface (Digitizer) möglich, das die einzelnen Klänge in digitalisierter Form an den Amiga weiterleitet. Insgesamt stehen vier Tonkanäle zur Verfügung. Um sich bei Musikfans Gehör zu verschaffen, werden jeweils zwei von ihnen zu einem Stereokanal zusammengefaßt.

Neben DMA und Sound ist in Paula noch die Interrupt-Verwaltung untergebracht, durch die der Amiga »multitaskingfähig« wird. Das bedeutet, daß mehrere Programme scheinbar gleichzeitig abgearbeitet werden. So kann zum Beispiel parallel ein Text gedruckt, eine Datei sortiert, ein Brief geschrieben und ein Musikstück angehört werden.

Der zweite Spezialchip im Amiga hat den wohlklingenden Namen »Denise« und ist für die Verwaltung der Grafik zuständig, ein Gebiet, das gerade der

Amiga im Gegensatz zu anderen Heim- und Personal Computern besonders gut beherrscht. (Bild 1 veranschaulicht, mit welcher hohen Auflösung Bilder auf dem Amiga dargestellt werden können.) Die der deutschen Fernsehnorm angepaßten PAL-Amigas bieten grundsätzlich eine Grafikauflösung von 320 x 256 Punkten in 32 beziehungsweise 640 x 256 Punkten in 16 Farben. Es existieren aber noch zwei weitere Modi: »Interlace« und »Hold Modify« (H.A.M.). Im ersten, dem Interlace-Modus, wird prinzipiell nichts anderes gemacht, als die Anzahl der horizontalen Zeilen bei halber Bildwechselfrequenz zu verdoppeln. Durch diese Eigenschaft erhält man zusätzlich zwei weitere Auflösungen bei PAL-Geräten (320 x 512 und 640 x 512 Punkte). Allerdings sinkt die Bildwechselfrequenz von 50 auf 25 Hz. Bemerkbar macht sich dies durch ein Flackern des Bildes. Der zweite, der H.A.M.-Modus, gestattet 4096 Farben in einem Bild zu verwenden. Dieser Modus ist in erster Linie dazu gedacht, Farben fließend ineinander übergehen zu lassen.

Denise kann aber noch mehr: Sogenannte »Play-Fields« lassen die Definition farbiger Spielfelder zu. Dabei kann ein solches Spielfeld eine annähernd unbegrenzte Größe annehmen. Vom Spielfeld selbst ist aber meist nur ein Ausschnitt in der gewählten Auflösung sichtbar. Dieser Ausschnitt läßt sich ähnlich einer Lupe auf einer Landkarte beliebig verschieben. Es besteht sogar die Möglichkeit, zwei solcher Play-Fields übereinander zu lagern, also Hires-Scrollen in zwei Ebenen zu realisieren. Allerdings verringert sich dadurch die Anzahl der Farben pro Play-Field auf acht. Der Ausschnitt, der gerade zu sehen ist, kann außerdem noch in der Größe variiert werden. Er kann so groß wie der Bildschirm oder so klein wie ein Buchstabe sein.

Zusätzlich kennen die Amiga-Modelle noch die schon vom C 64 her bekannten Sprites, bei denen es sich um kleine, frei verschiebbare Speicherblöcke handelt, in die eine Hires-Grafik eingeblendet werden kann. Insgesamt verfügt der Amiga über acht solcher Sprites mit einer Auflösung von 16 Punkten horizontal und einer frei definierbaren vertikalen Auflösung. Die Anzahl der Farben pro Sprite ist auf vier



Bild 1. Dieses Bild zeigt sehr deutlich, zu welchen Bildschirmauflösungen der Amiga in der Lage ist

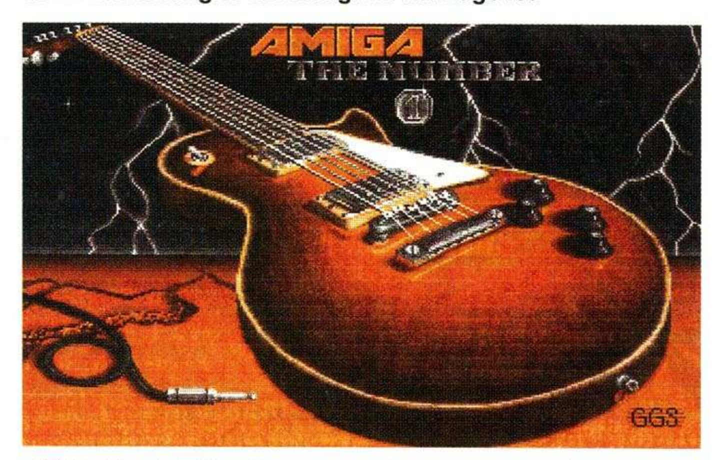


Bild 2. Eine Farbhardcopy, von einem Amiga und einem 24-Nadel-Farbdrucker (NEC CP6) erzeugt

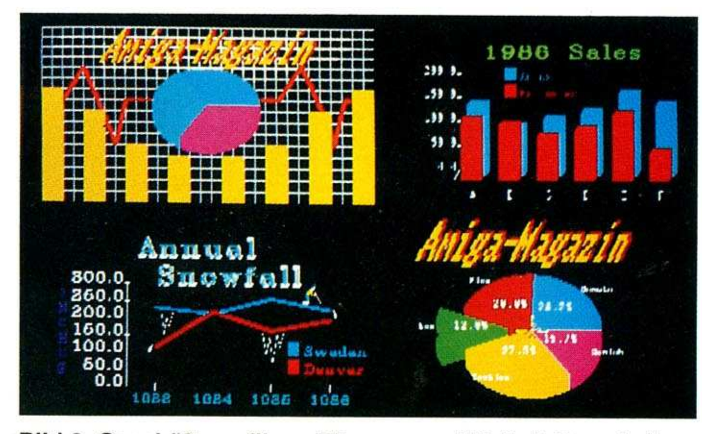


Bild 3. Geschäftsgrafiken, Bilanzen und Kalkulation sind mit der geeigneten Software kein Problem

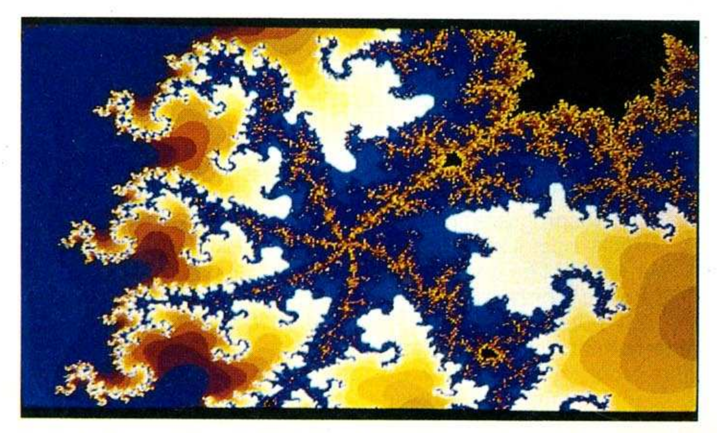


Bild 4. Dieses Mandelbrot-Bild wurde mit einem Programm erzeugt, das die Möglichkeiten des Amiga gut ausnutzt

begrenzt. Reichen einem diese vier Farben nicht aus, so kann man zwei Sprites übereinanderlegen, und erhält so ein Sprite mit erweiterter Farbpalette (16 Farben). Jedes dieser Sprites kann mehrmals auf dem Bildschirm dargestellt werden, sofern mindestens ein Rasterstrahl Abstand zwischen diesen ist. Natürlich kann der Amiga, ähnlich wie der C 64, Berührungen zwischen zwei Sprites feststellen und Prioritäten ändern, falls sich zwei Sprites überlagern. Der Programmierer kann also entscheiden, welches Sprite von welchem verdeckt ist.

Der dritte Spezialprozessor, der im Amiga seine Dienste verrichtet, trägt den Namen »Agnus«. Ihm obliegt ebenfalls ein Teil der Kontrolle über die Grafik wie beispielsweise Erkennung eines Rasterinterrupts, oder Verschieben großer Datenmengen im Speicher. Außerdem sorgt er für den Refresh der RAM-Bausteine, ohne den diese ihren Inhalt nicht behalten könnten.

Die Steuerung der Schnittstellen übernehmen zwei Ein-/Ausgabe-Bausteine vom Typ 8520. Sie werden vom System voll ausgelastet und können nicht für andere Zwecke eingesetzt werden.

Benutzeroberfläche »eingebaut«

Der Amiga wird normalerweise nicht mehr, wie von vielen Computern gewohnt, mit der Tastatur bedient. Vielmehr verfügt er über eine grafische Benutzerfläche (»Intuition«), in der mit Hilfe der mitgelieferten Maus (die Maus ist ein Eingabegerät ähnlich einem Joystick) die gewünschte Operation ausgewählt und aktiviert werden kann. Dazu müssen nur die gewünschten Funktionen oder Programme mit dem Mauszeiger (der »Cursor«) erfaßt werden. Ein ein- oder zweimaliger Druck auf eine Maustaste aktiviert dann die Ausführung. Dadurch muß der Benutzer nicht mehr viel Wissen über den Computer besitzen, um ihn bedienen zu können.

Anschlußmöglichkeit für viele Geräte

Interessant für jeden Anwender, der mit einem Amiga liebäugelt, ist die Frage nach den zusätzlich anschließbaren Pe-

ripheriegeräten. Betrachtet man die Rückseite des Computers, ist man erst einmal von der Anzahl der Stecker und Steckerleisten überrascht. Vorhanden ist eine Centronics-Schnittstelle, die den Anschluß handelsüblicher Drucker zuläßt, die ebenfalls eine Centronics-kompatible Schnittstelle besitzen. Die Hardcopy in Bild 2 wurde beispielsweise mit einem MPS 2000-Farbdrucker über den parallelen Port des Amiga ausgegeben.

Die bidirektionale Schnittstelle zur seriellen Datenübertragung (RS232-Standard) enthält alle erforderlichen Steuerleitungen. Über sie lassen sich Modems, Drucker und so weiter problemlos betreiben.

An die »Disk-Drive«-Schnittstelle können an jeden Amiga bis zu drei zusätzliche Diskettenlaufwerke (3½- und 5¼-Zoll) und Festplatten-Laufwerke mit zusätzlichem Controller angeschlossen werden.

Zum Monitor-Anschluß sind bei allen Amiga-Modellen RGB-Ausgänge vorhanden.

Der Amiga verfügt über einen Stereoausgang, der mit einem Verstärker oder einer vorhandenen Stereoanlage verbunden werden kann.

Es sind zwei Joystick-Ports eingebaut. Sie dienen neben dem Spieltrieb auch ernsthaften Aufgaben. So läßt sich zumindest an Port 1 sowohl eine Maus wie auch ein Lightpen oder ein Paddle anschließen. Bis auf den Lightpen ist Port 2 dem ersten gleichzusetzen. Allerdings wird er nicht vom Betriebssystem unterstützt. Das heißt, daß eine Maus oder ein Paddle im zweiten Port vom Anwenderprogramm aus abzufragen ist.

An Zusatzgeräten stehen den Amiga-Modellen inzwischen jede Menge zur Verfügung. Dies geht von Diskettenlaufwerken und Festplatten über Sprach- und Bilddigitalisierer und umfaßt verschiedenste Speichererweiterungen.

Außerdem ist jeder Amiga prinzipiell MS-DOS-fähig. (Dazu können »PCs« an das Bussystem angeschlossen werden, die den Zugriff auf die »Welt der PCs« ermöglichen.)

MS-DOS (MS-DOS = Microsoft-DOS = Diskettenbetriebssystem) ist ein hauptsächlich auf Personal Computern verbreitetes Betriebssystem, für das eine Unzahl von Programmen erhältlich ist. Seien es Textverarbeitungen, Datenbanken, Kalkulationsprogramme oder einfach nur Spiele.

Wie bei jedem Computer steckte die Software-Entwicklung anfangs noch in den Kinderschuhen. Inzwischen ist seit Erscheinen des ersten Amiga 1000 genug Zeit vergangen, in der die Programmierer und Entwickler nicht geschlafen haben. Wie die große Software-Marktübersicht in dieser Ausgabe zeigt, findet sich inzwischen für fast jede denkbare Anwendung das geeignete Programm. Seien es Spielprogramme, Programmiersprachen, Finanz- und Kalkulationsprogramme (Bild 3), Grafik- oder Lernsoftware. Außerdem gibt es derzeit über 200 Public Domain-Disketten, die mit unzähligen Programmen gefüllt sind, die Freaks und professionelle »Software-Designer« zum Selbstkostenpreis zur Verfügung gestellt haben. Das bedeutet, daß der Käufer dieser Disketten zu einem Preis von durchschnittlich 10 Mark auf Disketten zugreifen kann, die teilweise randvoll mit nützlichen Programmen und Routinen gefüllt sind.

Von Fall zu Fall läßt die Software zwar noch zu wünschen übrig, doch ist jetzt erst ein kleiner Schritt in der Ausnutzung des Amiga getan. Mit Sicherheit ist damit zu rechnen, daß immer mehr Programme entwickelt werden, die den Amiga und seine faszinierenden Eigenschaften so ausnutzen, wie es eigentlich sein sollte. Der Trend zu »maschinenorientierter« Software ist aber derzeit klar zu erkennen, da immer mehr Programme auf dem Markt erscheinen, die dem Amiga das abverlangen, was er zu geben bereit ist. Ein Beispiel dafür sind Apfelmännchen, anderen auch als Mandelbrot-Grafiken bekannt. Das Apfelmännchen in Bild 4 beispielsweise wurde mit einem Public Domain-Programm (!) erzeugt (MandFXP), das solche Berechnungen in relativ »kurzer« Zeit erledigt. Bedenkt man den Rechenaufwand, der beispielsweise (für Eingeweihte) bei 1000 Iterationen anfällt, erscheinen drei Stunden gewiß nicht zuviel.

Bild 5. Der Amiga 1000 mit einem Zusatzlaufwerk, RGB-Monitor und dem MS-DOS-Zusatz »Sidecar«. Dieser Zusatz ermöglicht den Zugriff auf »die Welt der PCs«.



Bild 6. Der neue Heimcomputer Amiga 500 präsentiert sich in angenehmen Design und (gegenüber dem Amiga 1000) aufgelockerter Tastatur mit abgesetztem Cursor-Block

Doch damit genug der allgemeinen Vorstellung der Amiga-Modelle. Widmen wir uns nun den einzelnen Geräten, wobei auf die Unterschiede der Modelle zueinander eingegangen wird. Zum Lieferumfang eines jeden Computers gehört eine Maus, deutschsprachige Handbücher sowie zwei Disketten, auf denen die Workbench mit der Benutzeroberfläche und das Amiga-Basic enthalten sind. Ein Monitor gehört bei keinem der Geräte zur Grundausstattung, ebensowenig wie ein Joystick oder ein MS-DOS-Zusatz.

Als erstes nehmen wir den Amiga 1000 unter die Lupe:

Das Ursprungsmodell der Amiga-Modellreihe, der Amiga 1000 (Bild 5), verfügt in der Grundausstattung über einen Hauptspeicher von 256 KByte RAM. Dieser läßt sich durch eine von Commodore gefertigte RAM-Erweiterung, die in die vordere Aussparung gesteckt wird, um 256 KByte auf insgesamt 512 KByte erweitern. Es werden jedoch von einigen Anbietern auch andere Speichererweiterungen angeboten, die den Hauptspeicher auf 4,5 MByte und mehr aufstocken können. Eine Erweiterung auf mindestens 1 MByte ist zu empfehlen, da manche Anwendungen nur mit einem großen Speicherplatz sinnvoll arbeiten können.

Auf dem Amiga 1000 sind uneingeschränkt alle Programme lauffähig, da sich bei ihm verschiedene Kickstart-Versionen per Diskette laden lassen. (Der Kickstart enthält das eigentliche Betriebssystem des Amiga.) Die derzeit aktuellste ist die Kickstart-Version 1.2. Es gibt aber einige Programme, die auf die ältere Kickstart-Version 1.1 angewiesen sind und aus diesem Grund nur mit dem Amiga 1000 laufen.

Der Amiga 1000 besitzt ein eingebautes 3½-Zoll-Diskettenlaufwerk sowie eine abgesetzte Tastatur, die sich bei Bedarf unter den Computer schieben läßt. Dies hilft auf Schreibtischen mit beengten Platzverhältnissen, den Freiraum zu vergrößern.

AMIGA Markt&Technik

PROGRAMM-SERVICE

Direkt bestellen statt abtippen!

Diskettenoperationen vereinfacht

Diesmal finden Sie auf unserer Programmservice-Diskette wieder viele interessante Programme. Mit »ExDir«, dem Programm des Monats, sind alle Schwierigkeiten mit den Subdirectories endlich vorbei. Mit »Hardcopy« drucken Sie Teile des Screens aus. Um Ihre Verfassung zu erfahren, verwenden Sie einfach »Bio«, ein fantastisches Biorhythmusprogramm. Einfaches Erzeugen auch von komplexen Eingabemasken wird mit »Mask« möglich. »Freeman« zeigt den verfügbaren Speicher im CLI-Fenster an. Zum Lernen von Vokabeln ist »Vokabel« ein sehr gutes Werkzeug mit viel Komfort. Wie man Sprites auf dem Amiga programmiert, sehen Sie in dem Demoprogramm »SimpleSprites«. Die Veränderung des CLI-Fensters stellt mit »InitCLI3« kein Problem mehr dar. Wenn Sie gerne spielen, ist »Vier gewinnt« genau das richtige Programm für Sie.

Best.-Nr. 48702 sFr 24,90/öS 299,-* **DM 29,90***

* inkl. MwSt. Unverbindliche Preisempfehlung

Leckerbissen für C-Programmierer

Netzgrafik in Amiga-Basic: Dreidimensionale Darstellung von Funktionen. **Disketten-Monitor:** Geschrieben in Amiga-Basic mit vielen Möglichkeiten. **Polydraw in Amiga-Basic:** Hier können Sie etwas lernen über die Benutzung der Grafikbibliothek. **FTYPE:** Drucken mit Komfort mit C-Programm. **FCat in C:** Schnelle Directory-Routine. **Font-Designer:** Amiga-Basic-Programm zum Erstellen von eigenen Fonts. **Menu-Routinen:** Benutzung der Betriebssystemroutinen von C zum Erzeugen von Pull-down-Menüs. **Objekt-Editor in Amiga-Basic:** BOB- und Spriteeditor mit komfortabler Maussteuerung. **InitCli:** C-Programm zur Veränderung von Cli-Fenstern, mit vielen Möglichkeiten wie zweifarbige Darstellung und Interface-Modus. (Aus Ausgabe 6/7-87).

Best.-Nr. 48701 sFr 24,90/öS 299,-* **DM 29,90***

* inkl. MwSt. Unverbindliche Preisempfehlung

Bestellungen bitte an: Markt&Technik Verlag AG, Unternehmensbereich Buchverlag, Hans-Pinsel-Straße 2, D-8013 Haar, Telefon (089) 4613-0. **Schweiz:** Markt&Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Telefon (042) 41 56 56. **Österreich:** Ueberreuter Media Handels- und Verlagsgesellschaft mbH (Großhandel), Alser Straße 24, A-1091 Wien, Telefon (0222) 48 15 38-0, Microcomputique E. Schiller, Fasangasse 24, A-1030 Wien, Telefon (0222) 78 56 61, Bücherzentrum Meidling, Schönbrunner Straße 261, A-1120 Wien, Telefon (0222) 83 31 96. **Bestellungen aus anderen Ländern bitte nur schriftlich an:** Markt&Technik Verlag AG, Abt. Buchvertrieb, Hans-Pinsel-Straße 2, D-8013 Haar, und gegen Bezahlung einer Rechnung im voraus.

Bitte verwenden Sie für Ihre Bestellung und Überweisung die abgedruckte Postgiro-Zahlkarte, oder senden Sie uns einen Verrechnungs-Scheck mit Ihrer Bestellung. Sie erleichtern uns die Auftragsabwicklung, und dafür berechnen wir Ihnen keine Versandkosten.

Einlieferungsschein/Lastschriftzettel
(nicht zu Mitteilungen an den Empfänger benutzen)

Gebühr für die Zahlkarte
(wird bei der Einlieferung bar erhoben)
bis 10 DM ——— 90 Pf
über 10 DM (unbeschränkt) 1,50 DM
Bei Verwendung als Postüberweisung gebührenfrei

Bedienen Sie sich der Vorteile eines eigenen Postgirokontos
Auskunft hierüber erteilt jedes Postamt

Feld für postdienstliche Zwecke

Hinweis für Postgirokontoinhaber:
Dieses Formblatt können Sie auch als Postüberweisung benutzen, wenn Sie die stark umrandeten Felder zusätzlich ausfüllen. Die Wiederholung des Betrages in Buchstaben ist dann nicht erforderlich. Ihren Absender (mit Postleitzahl) brauchen Sie nur auf dem linken Abschnitt anzugeben.
1. Abkürzung für den Namen Ihres Postgiroamts (PGiroA) siehe unten
2. Im Feld »Postgiroteilnehmer« genügt Ihre Namensangabe
3. Die Unterschrift muß mit der beim Postgiroamt hinterlegten Unterschriftsprobe übereinstimmen
4. Bei Einsendung an das Postgiroamt bitte den Lastschriftzettel nach hinten umschlagen

Abkürzungen für die Ortsnamen der PGiroÄ:

Bln W = Berlin West, Kln = Köln
Dtmd = Dortmund, Lshfn = Ludwigshafen am Rhein
Esn = Essen
Ffm = Frankfurt am Main, Mchn = München
Nbg = Nürnberg
Hmb = Hamburg, Sbr = Saarbrücken
Han = Hannover, Stgt = Stuttgart
Klrh = Karlsruhe

Für Mitteilungen an den Empfänger

Bestellung Programm-Service **Wichtig:** Lieferanschrift (Rückseite) nicht vergessen!

Bestell-Nr.	Anzahl	x Einzelpreis	= Gesamtpreis
Summe bitte auf Vorderseite übertragen		Gesamtsumme:	

Im Gegensatz zu den anderen Modellen kann außer dem RGB-Monitor auch ein Composite-Monitor oder Videorecorder angeschlossen werden.

Der Anschluß eines Druckers an die Centronics-Schnittstelle ist dagegen nicht so ohne weiteres zu bewerkstelligen. Leider hat Commodore bei diesem Gerät an Pin 23 der Parallel-Buchse +5 V angeschlossen, was unter Umständen zu einer Zerstörung eines angeschlossenen Druckers führen kann, wenn diese Leitung im Druckerkabel nicht durchtrennt wird. Hierauf sollte beim Kauf eines Druckerkabels besonderes Augenmerk gerichtet werden!

An den Amiga 1000 kann ein weiterer Zusatzcomputer, der »Sidecar«, angeschlossen werden, der dem Anwender die MS-DOS-Welt eröffnet (Bild 5). Im Prinzip ist der Sidecar ein Personal Computer (PC), bei dem allerdings alle für die Bildschirmausgabe wichtigen Teile fehlen. Die monochrome und farbige Darstellung der PC-Grafik emuliert und übernimmt dabei der Amiga.

Der Sidecar, der mit einem 8088-Prozessor bestückt ist, besitzt in der Grundausstattung einen Hauptspeicher von 256 KByte (aufrüstbar auf 512 KByte) und verfügt über ein eingebautes 5¼-Zoll-Diskettenlaufwerk. Ein freier Sockel für den Einbau eines 8087-Arithmetikprozessors ist vorhanden (der 8087 verhilft zu Geschwindigkeitssteigerungen, wenn mathematische Berechnungen durchgeführt werden). Die Taktfrequenz des PC-Systems beträgt 4,77 MHz und ist nicht steigerbar (etwa auf 8 MHz). Ausführliche Benchmark-Tests (Geschwindigkeitstests) ergaben einen Norton-Faktor (»standardisierter« Wert zur Messung von Rechengeschwindigkeit, um einen Vergleichswert zu haben) von 0,98 bis 1,0. Das bedeutet, daß der Sidecar genauso schnell ist wie ein Original-IBM-PC. Lediglich bei der Bildschirmausgabe verliert er an Geschwindigkeit, da der Amiga die Grafikdarstellung übernimmt. Abhilfe schafft hier beispielsweise der Einbau einer Hercules-, EGA- oder CGA-Grafikkarte in einen der insgesamt drei freien PC-kompatiblen Steckplätze.

Ein extern mit dem Amiga verbundenes 3½-Zoll-Diskettenlaufwerk kann vom Sidecar aus benutzt werden. Selbstverständlich können auch 5¼-Zoll-Laufwerke oder Festplatten an den Amiga 1000 oder den Sidecar angeschlossen werden.

Probleme oder Inkompatibilitäten ergaben sich unter MS-DOS praktisch keine. Bis jetzt traten mit allen getesteten Standardprogrammen keine Schwierigkeiten auf.

Fazit Amiga 1000

Der Amiga 1000 kann als preiswertes Einsteigermodell angesehen werden. Gerade jetzt, da die »neuen« Amiga-Modelle auf den Markt kommen, ist dieses Gerät zu relativ »günstigen« Preisen zu erstehen.

Vorteilhaft ist, daß fast alle derzeit angebotenen Hardware-Erweiterungen an den Amiga 1000 passen, da sie speziell für ihn entwickelt wurden (beispielsweise Sidecar). Auch kann man derzeit sicher sein, daß alle Programme auf ihm laufen, was bei den anderen Modellen leider nicht der Fall ist.

A500: neuer Heimcomputer?

Das preiswerteste der drei Modelle ist der Amiga 500 (Bild 6), dessen Zielgruppe der Heimcomputer-Markt ist. Er ist optisch eine Verbindung des C 128 und des Atari, also ein kompaktes Gerät, bei dem die (sehr angenehm zu bearbeitende) Tastatur und der eigentliche Computer in einem Gehäuse vereint sind.

In der Grundausstattung besitzt der Amiga 500 ein eingebautes 3½-Zoll-Diskettenlaufwerk und 512 KByte Hauptspeicher (durch eine zusätzliche Einsteckkarte mit integrierter, batteriegepufferter Echtzeituhr auf insgesamt 1 MByte intern und bis zu über 8 MByte extern erweiterbar). Das Betriebssystem (Kickstart-Version 1.2) muß nicht mehr von Diskette geladen werden, sondern ist bereits in einem ROM-Baustein fest eingebaut. Dies hat leider den Nachteil, daß Programme, die die Kickstart-Version 1.1 benötigen, nicht mehr uneingeschränkt oder überhaupt nicht mehr funktionieren. Derzeit dürften aber nahezu alle Software-Hersteller an Versionen arbeiten, die auf allen drei Systemen funktionieren.

Ebenfalls kommt es bei manchen Programmen zu Schwierigkeiten, wenn die Uhrenkarte eingesteckt ist. Mit einigen Kniffen (wie etwa das Entfernen der Speichererweiterungs-Karte) und einem mitgelieferten Hilfsprogramm kann aber doch ein Großteil der Software zum Laufen gebracht werden.

Ein erfreulicher Aspekt ist, daß beim Amiga 500 endlich alle Ports normgerecht sind, das heißt, daß der parallele Port nicht mehr auf Pin 23 die positive Spannung führt.

Einige Erweiterungen des Amiga 1000 wie beispielsweise der Sidecar lassen sich mit einigen Tricks (der Expansion-Port des Amiga 500 ist um 180 Grad gedreht) auch für den »kleinen« Amiga verwenden. Solange es um Zusätze geht, die am Expansion-Port angeschlossen werden, können diese mit einem Adapterkabel, das den »verdrehten« Port berücksichtigt, verbunden werden.

Leider kann nur ein analoger RGB-Monitor direkt angeschlossen werden. Stecker für Videorecorder oder Composite-Monitore sind nicht mehr vorhanden. Es ist aber für etwa 60 Mark ein Adapter erhältlich, mit dem (für den schmalen Geldbeutel) ein Fernsehgerät angeschlossen werden kann.

Fazit Amiga 500

Das preiswerteste der drei Amiga-Modelle, der Amiga 500, wird sich sicher in Zukunft im Heimcomputer-Markt behaupten, wenn nicht durchsetzen können. Leider ist er nicht vollständig softwarekompati-



bel zum Amiga 1000, das heißt, daß auf dem Amiga 500 derzeit nicht alle Programme funktionieren. Mit einigen Tricks kann jedoch der größte Teil der Programme doch zum Laufen gebracht werden. Es ist aber zu erwarten, daß alle Programme, die jetzt produziert werden, kompatibel sind.

Der Amiga 500 besitzt eine ins Gehäuse integrierte Tastatur, mit der sich sehr gut arbeiten läßt. Die Tastatur verfügt zusätzlich zu der des Amiga 1000 über einen separaten Cursor-Block und ist auch etwas »aufgelockerter«, die Tasten liegen nicht mehr so dicht beieinander wie bei der Amiga 1000-Tastatur.

Es lassen sich auch über ein entsprechendes Adapterkabel die meisten, für den Expansions-Bus des Amiga 1000 konzipierten Geräte anschließen (auch der Sidecar).

Leider muß die Möglichkeit, einen Videorecorder oder Composite-Monitor anschließen zu können, extra bezahlt werden. Da der Anschluß eines Composite-Monitors bei der Grafikauflösung des Amiga aber nicht zu einem optimalen Ergebnis führt, kann darüber getrost hinweggesehen werden.

Für Profis

Die absolute Traummaschine der Freaks und kommerziellen Anwender ist wohl der Amiga 2000 (Bild 7). Er besitzt ein formschönes PC-Gehäuse mit abgesetzter Tastatur, die baugleich zu der des A 500 ist.

In der Grundausstattung verfügt er über 1 MByte RAM, eine batteriegepufferte Echtzeituhr und ein 3½-Zoll-Diskettenlaufwerk.

Zusätzlich läßt sich der interne Hauptspeicher auf bis zu 8,5 MByte aufrüsten. Ein zweites 3½-Zoll-Laufwerk kann intern eingebaut werden.

Als besonderen Leckerbissen bietet Commodore einen kompletten PC als Steckkarte zum Einbau in die insgesamt vier verfügbaren PC/AT-Steckplätze an (Bild 8). (Einen Steckplatz belegt dabei die PC/XT-Karte.) Dazu wird ein 5¼-Zoll-Laufwerk geliefert, das ebenfalls von vorne zugänglich installiert werden kann. Dieses Laufwerk läßt sich auch vom Amiga aus nutzen, genauso, wie der PC eines der 3½-Zoll-Laufwerke ansprechen kann. Die PC/XT-Platine ist mit einem 8088-Prozessor bestückt, der mit 4,77 MHz getaktet wird. Ein freier Steckplatz für einen 8087-Arithmetik-Prozessor ist vorhanden. Die PC-Karte wird mit 512 KByte Speicher ausgeliefert. Geschwindigkeitstests ergaben einen Norton-Faktor (Erklärung siehe Amiga 1000/ Sidecar) von 0,98 bis 1,0. Er ist also genauso schnell wie ein original IBM-PC. Lediglich bei der Bildschirmausgabe müssen Geschwindigkeitseinbußen in Kauf genommen werden, da der Amiga die Bildschirmausgabe des PC (monochrom und CGA-Modus mit 16 Farben) übernimmt. Abhilfe schafft hier eine Grafikkarte für den PC, der in einen der freien Slots eingesteckt wird. In unser Testgerät wurde eine CGA-Karte eingebaut, die reibungslos funktionierte.

Commodore hat auch eine 20-MByte-Festplatte mit Controller für den Amiga 2000 angekündigt, ebenso wie eine PAL-Video-Karte, die den Anschluß von Composite-Monitoren oder Videorecordern ermöglichen soll. Unser Testgerät wurde aber zu Kompatibilitätstests mit zwei 20-MByte-Festplatten von Seagate und einem OMTI-Controller bestückt.

Der Amiga 2000 besitzt des weiteren noch fünf Amiga-kompatible Steckplätze, die für die Aufnahme von Amiga-spezifischen Karten vorgesehen sind. Davon wird ein Slot für die Aufnahme der PC/XT- oder AT-Karte benötigt. Ebenfalls soll der, nach Aussagen von Commodore, bald erhältliche Festplatten-Controller in einen dieser Amiga-Slots eingesteckt werden können.



Bild 7. Das Profigerät: Ein zum PC aufgerüsteter Amiga 2000 mit drei Diskettenlaufwerken und RGB-Monitor

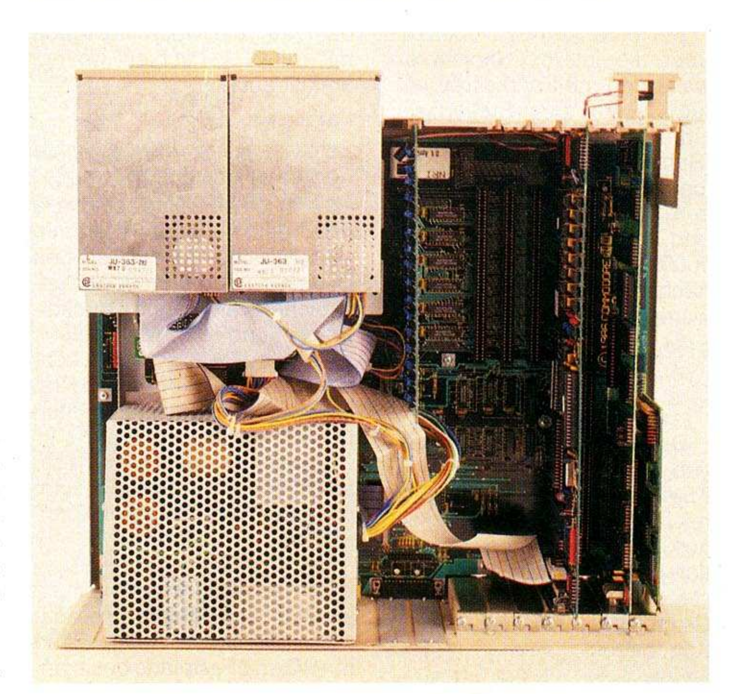


Bild 8. Ein Blick ins Innere des Amiga 2000. Deutlich sichtbar ist der MS-DOS-Teil (rechts) mit PC-Karte, Festplattencontroller und CGA-Grafikkarte, die reibungslos funktioniert.

Laut Ankündigung von Commodore soll es auch bald eine AT-Karte geben, die voraussichtlich mit einem 80286-Prozessor ausgerüstet sein wird und volle Kompatibilität aufweisen soll. Dazu dürfte dann natürlich auch ein 5¼-Zoll-Diskettenlaufwerk mit 1,2 MByte geliefert werden. Auch eine Turbo Amiga-Karte mit einem 68020 Prozessor soll, laut ITC, bald erhältlich sein.

Der Amiga 2000 besitzt, wie der Amiga 500, das Betriebssystem (Kickstart-Version 1.2) im ROM (ROM = nichtflüchtiger Speicher). Auch auf diesem Amiga laufen leider nicht alle Programme. (Es gilt das beim Amiga 500 Gesagte.)

Fazit Amiga 2000

Der Amiga 2000 ist derzeit wohl eine Traummaschine für jeden Computerinteressierten, sei es Freak oder ernsthafter Anwender. Er ist ein offenes System, das heißt, er kann vielfältig aufgerüstet werden. Durch seine PC/XT/AT-Steckplätze steht ihm die ganze MS-DOS-Welt und später vielleicht sogar die Unix-Welt offen. Dies macht sich natürlich im Preis bemerkbar: Für einen Amiga 2000, der mit PC/XT-Karte, drei Diskettenlaufwerken, einem RGB-Monitor, mehr Speicher und einer Festplatte »voll« aufgerüstet ist, kann man derzeit ab 7000 Mark aufwärts rechnen. Dem Preis sind nach oben hin fast keine Grenzen gesetzt.

Der Amiga 2000 bietet sich jedem an, der den Computer im professionellen Bereich nutzen will und auch mal Zeit für ein kleines Spielchen mit einer bewundernswerten Grafik übrig hat.

Diese »kleine« Vorstellung der drei Amiga-Modelle wird Ihnen etwas bei der Auswahl des für Sie am besten geeigneten Amiga geholfen haben. Jedes der Modelle hat Vor- und Nachteile: Sei es der Preis, das Leistungsvermögen oder die Kompatibilität zu bestehender Software.

Was bleibt, ist auf jeden Fall der Spaß beim Träumen vom neuen Amiga, der hoffentlich bald auf Ihrem Tisch stehen wird... (dm)

Frühjahrsputz für die Workbench-Disketten

Die beiden mitgelieferten Disketten des Amiga enthalten allerlei Programme. Doch wissen Sie, was dies für Programme sind und welche man entbehren kann?

Zu jedem Amiga neueren Datums werden zwei Disketten mitgeliefert, die die Workbench V1.2 und das Amiga-Basic V1.2 sowie Extra-Programme enthalten. Es sind jedoch nicht alle Programme zum Betrieb des Computers nötig. Früher oder später möchte man gerne andere Programme auf die Arbeits-Workbench kopieren. Doch woher soll man den Platz dafür nehmen? Es müssen also Programme gelöscht werden. Wir sagen Ihnen, wie Sie Ihre Arbeitsdiskette »säubern« können. (Sie sollten aber einmal alle Programme ausprobiert haben, um wirklich zu wissen, ob sie nicht doch noch von Ihnen benötigt werden.)

Arbeiten Sie aber nur mit Kopien der Originaldisketten!

Die Säuberungsaktion

Sobald Sie dies erledigt haben, können wir beginnen. Legen Sie dazu bitte die Workbench-Diskette ein und aktivieren das CLI (Command Line Interface) im System-Ordner oder besser, ein Hilfsprogramm wie Zing! oder CLI-Mate. Wir werden nun der Reihe nach ein Subdirectory nach dem anderen durchgehen und erklären, was es mit den Dateien auf sich hat. Sollten Sie einige Dateien entdecken, die auf Ihren Disketten nicht enthalten sind, können wir Sie beruhigen: Es gibt laufend neue Versionen der Workbench, auf die je nach Anforderung neue Programme kopiert wurden. Falls Ihr Systemhändler eine Kopie der neuesten Version (derzeit V33.56) besitzt, wird er sich sicher gerne bereit erklären, Ihnen eine Kopie zu erstellen.

Beginnen wir gleich mit der Arbeit. Das erste Verzeichnis, das wir uns vornehmen wollen, ist das Directory »c« auf der Workbench-Diskette. Dieses Verzeichnis enthält alle Befehle des CLI (wenn Sie später neue CLI-Kommandos bekommen, müssen Sie diese in den »c«-Ordner kopieren). Begeben Sie sich mit dem Befehl

```
CD c
```

in diesen Ordner. Der Befehl

```
DIR
```

zeigt Ihnen die einzelnen Programme in diesem Ordner an. Hier nun eine Liste der Programme, die leicht entbehrlich sind und mit dem Befehl

```
DELETE Name
```

gelöscht werden können. Falls Sie über einen Amiga 2000 mit Festplatte verfügen, können Sie die nun folgenden Files entfernen (vergewissern Sie sich bitte erst durch ein Studium des DOS-Handbuchs, ob Sie wirklich auf diese wenig gebrauchten Befehle verzichten wollen):

```
Ask
Break
ChangeTaskPri
Edit
Fault
Join
Lab
Quit
SetDate
Sort
Status
Wait
Why
```

Bei einem Amiga 500, der über die interne Speichererweiterung mit Echtzeituhr verfügt, sollte zum Stellen der Uhr der Befehl »SetClock« (Schreibweise: SetClock opt load) behalten werden. Besitzen Sie die Echtzeituhr nicht und einen Amiga 500 oder 1000, kann zusätzlich auf die nächsten Befehle verzichtet werden:

```
DiskChange
DJMount
SetClock
```

Durch diese Säuberungsaktion haben Sie sich etwa 45 KByte Platz geschaffen. Dies reicht aber meist noch nicht aus. Deshalb setzen wir die Aktion gleich im nächsten Unterverzeichnis fort: dem »Devs«-Ordner. Wechseln Sie mit dem Kommando

```
CD /Devs
```

auf diese Ebene über.

Das (leere) Subdirectory »ClipBoards« können Sie entfernen, wenn Sie keinen Amiga 2000 mit PC- oder AT-Karte oder den Sidecar besitzen.

Im nächsten »Devs«-Ordner »KeyMaps« sind alle Tastaturtreiber unnötig, die nicht zu unserem Standard passen. Es sind dies alle Dateien außer »d« für die deutsche oder »usa1« für die amerikanische Tastaturbelegung. Entfernen Sie die nicht benötigten Treiber aus diesem Directory.

Das Directory »Printers« enthält viele Druckertreiber. Löschen Sie dort alle Druckertreiber außer dem, den Sie im Preferences-Menü eingestellt haben (etwa Epson oder MPS-2xxx).

Nach den Subdirectories folgen einige Gerätetreiber, die durch die Endung ».device« erkannt werden können. Sollten Sie nicht über einen Amiga 2000 mit PC- oder AT-Karte oder einen Sidecar verfügen, sind die Dateien

```
Clipboard.device
JDisk.device
```

überflüssig. Wenn Sie außerdem auf die Sprachausgabe mit »Say« verzichten wollen, entfällt auch

```
Narrator.device
```

Die Datei »System-Configuration« ist zwingend erforderlich. In dieser Datei sind alle mit dem Setup-Programm »Preferences« voreingestellten Daten und Konfigurationen gespeichert, also auch Druckertreiber und der Mauszeiger. Wird diese Datei gelöscht, können also nicht mehr die in Preferences eingestellten Daten benutzt werden.

Damit können wir ins nächste Directory überwechseln: das Unterverzeichnis »Expansion«.

Die Dateien »Janus.library« und »Janus.library.info« müssen nur dann behalten werden, wenn Sie einen Amiga 2000 mit PC und Festplatte besitzen. Andernfalls ist das ganze Directory überflüssig und darf entfernt werden.

Da der Amiga den Standard-Zeichensatz fest im Kickstart eingebaut hat, ist das Directory »Fonts« (= Zeichensätze) nur reine Beigabe. Wenn Sie die Zeichensätze entbehren können, steht einer Löschung des gesamten Directorys (fast) nichts mehr im Wege. Es sei denn, Sie besitzen einen Amiga 2000 mit PC- oder AT-Karte oder einen Sidecar. Dann können alle Einträge (auch Subdirectories) in diesem Verzeichnis außer dem Subdirectory »PC.FONT« und der Datei »pc.font« gelöscht werden.

Bis jetzt dürfte auf Ihrer Diskette bereits ein merklicher Freiraum entstanden sein, den Sie mit für Ihre Anwendungen geeigneteren Programmen auffüllen können. Doch wir wollen keine halben Sachen machen. Deshalb gehen wir über zum nächsten Subdirectory, dem Verzeichnis »l«.

Die Diskette wird »freier«

Der Aux-Handler wird benötigt, wenn über die serielle Schnittstelle beispielsweise Daten vom CLI aus gesendet werden. Man kann ihn aber vorerst entbehren, da diese Anwendung tiefere Kenntnisse von der Hard- und Software des Amiga verlangt.

Die Datei Disk-Validator sollte auf alle Fälle im »l«-Ordner belassen werden, da sie beim Arbeiten mit »DiskDoctor« benötigt wird.

Falls an Ihrem Amiga kein Drucker, Modem oder anderes Gerät an die parallele oder serielle Schnittstelle angeschlossen ist, sind die Files »Port-Handler« und »Pipe-Handler« entbehrlich. Soll nicht mit einer RAM-Floppy gearbeitet werden, kann der »Ram-Handler« auch gleich gelöscht werden.

Damit schwenken wir um in das Unterverzeichnis »Libs«, welches für die Systemverwaltung des Amiga wichtige Dateien enthält. Keinesfalls sollten Sie folgende Files löschen:

```
diskfont.library
icon.library
info.library
version.library
```

Falls nicht beabsichtigt ist, mittels des »Say«-Befehls Sprachausgabe laufen zu lassen, verzichtet man auf »translator.library«. Die Bibliotheken »mathieeedoubbas.library« und »mathtrans.library« werden in den allerwenigsten Fällen gebraucht. Sie stellen nur Dateien dar, auf die spezielle Routinen mit mathematischer Unterstützung zugreifen. Sie dürfen also entfernt werden.

Sollten Sie über einen Amiga mit Sidecar oder sogar über einen Amiga 2000 mit PC- oder AT-Karte verfügen, ist der Ordner »PC« für Sie tabu. Diese darin enthaltenen Programme benötigt das MS-DOS-System, um den PC-Bildschirm zu emulieren und den Kontakt zwischen den beiden Systemen zu ermöglichen. Besitzen Sie dagegen keines dieser MS-DOS-Geräte, steht es Ihnen frei, das gesamte Unterverzeichnis zu löschen. Das eben Gesagte gilt ebenso für das Dateiverzeichnis »Sidecar«. Auch hier sollten MS-DOS-Anwender auf dem Amiga keine Löschungen vornehmen.

Allgemein ist zu beachten: Das Directory »s« ist ebenfalls tabu, da dort die Boot-Datei steht, also die Datei, die die einzelnen Programme beim Starten lädt.

Im »System«-Ordner befindet sich dagegen wieder eine ganze Menge Software, die nicht oder nur selten benötigt wird. Entfernen können Sie aus diesem Verzeichnis die nachstehenden Files:

```
GraphicDump
GraphicDump.info
IconEd
IconEd.info
InitPrinter
InitPrinter.info
```

Die beiden »NoFastMem«-Dateien sollten nur entfernt werden, wenn Sie einen Amiga 1000 Ihr eigen nennen. Bei den beiden anderen Amiga-Typen sind diese Programme öfter nützlich, wenn es darum geht, Software, die komischerweise nicht läuft, doch zum Funktionieren zu bringen.

Damit gelangen wir zum nächsten Ordner, der den Namen »t« trägt. Er ist nur für temporäre Dateien da und kann ersatzlos entfernt werden. Ebenso verhält es sich mit »trashcan«. Wenn Sie den Abfalleimer nicht unbedingt benötigen, kann man auch dieses Unterverzeichnis löschen.

Das letzte Verzeichnis, »Utilities«, enthält in den neueren Versionen der Workbench nur noch das »Notepad«. Hier hängt es von Ihnen ab, ob Sie mit dem kleinen Texteditor arbeiten oder nicht. Wenn dies nicht der Fall ist, darf auch dieses letzte Unterverzeichnis ersatzlos gelöscht werden.

Begeben wir uns wieder mit »CD :« zurück in die oberste Directory-Ebene der Diskette. Dort finden sich noch ein paar Dateien, die teilweise nicht mehr nötig sein werden. Grundsätzlich dürfen Sie nun alle »Name.info«-Dateien entfernen, zu denen kein passendes Gegenstück (»Name«) mehr vorhanden ist. Beispielsweise kann »Trashcan.info« gelöscht werden, wenn der Ordner »Trashcan« nicht mehr existiert. Nur die Files »Disk.info«, ».info«, »Preferences« und »Preferences.info« sollten nicht entfernt werden.

Anders verhält es sich mit den beiden »Clock«-Dateien, die in den neueren Workbench-Versionen ihren Platz im Hauptdirectory gefunden haben. Wird die Uhr nicht benötigt, können auch die beiden »Clock«-Dateien entfallen.

Die Workbench-Diskette ist schlußendlich um 200 bis 300 KByte »leichter« geworden. Diesen Platz können Sie nun gewinnbringend nutzen, indem Sie wertvolle Utilities wie beispielsweise die »Metacomco Shell« oder »CLI-Mate« darauf unterbringen.

Auf der zweiten Diskette, der Extra-Diskette, sind eigentlich außer Amiga-Basic im großen und ganzen nur Demoprogramme enthalten. Hier müssen Sie probieren, auf welche dieser Routinen Sie verzichten möchten. Prizipiell ist es aber nicht nötig, Dateien von dieser Diskette zu löschen, da sie im Regelfall nicht als System- oder Arbeitsdiskette benutzt wird.

Durch diese Hinweise und Tips besser informiert, haben Sie nun hoffentlich das Problem der vollen Workbench-Diskette überwunden und können neue, nutzbringende Programme zum effektiven Arbeiten auf ihr unterbringen.

(dm)



Das können AMIGA 500 & 2000

Für den Einsteiger, der mehr über den Amiga wissen will, bietet sich dieses Buch an. Durch ein neues Konzept findet der Leser leicht die Stellen im Buch, die ihn interessieren. Auf den ersten zehn Seiten sind die einzelnen Kapitel durch Kurzbeschreibungen vertreten, in denen nur das Wichtigste steht. Diese Erklärungen sind in zwei Spalten aufgeteilt, wobei links die Erläuterung für Einsteiger und rechts die für Profis steht. Diese Seiten können später zum Nachschlagen verwendet werden, aber der eigentliche Sinn liegt in den Seitenangaben am Ende der Spalte, die auf die ausführlichen Kapitel verweisen.

In den folgenden Abschnitten werden nun die Fähigkeiten des Amiga erläutert, ohne auf dessen Programmierung einzugehen. Durch die zahlreichen Hinweise und Beschreibungen von vorhandenen Programmen zeichnet sich das Buch als Kaufhilfe aus, die noch Unentschlossene unterstützen soll. Auch die vielen Bilder sind eine gute Hilfestellung bei der Auswahl des richtigen Programms.

Von Anwenderprogrammen für Grafik, Animation und Musik über Programmiersprachen bis zu Hardware-Erweiterung findet der Leser alles, was ihm bei seiner Kaufentscheidung helfen kann.

Alles in allem ein Buch, das für denjenigen geeignet ist, der sich einen Computer kaufen will und noch nicht recht weiß, ob der Amiga der richtige Computer für ihn ist oder nicht.

(rb)

Stein, Das können AMIGA 500 & 2000, Data Becker GmbH, 190 Seiten, ISBN 3-89011-178-5, 29 Mark



Amiga Basic

Basic ist noch immer einer der schnellsten und einfachsten Wege zur Programmierung. Doch Umsteiger von anderen Heimcomputern werden merken, daß Microsoft Basic für den Amiga leistungsfähiger ist. Und da das von Commodore mitgelieferte Basic-Handbuch eigentlich nur weiter fortgeschrittenen Basic-Programmierern von Nutzen ist, wünscht sich der Anfänger ein verständliches Buch, das ihn Schritt für Schritt mit diesem Basic-Dialekt vertraut macht. Das sehr umfangreiche Buch enthält fast alles, was Einsteiger und Aufsteiger wissen sollten. Das Spektrum reicht von Grafik bis hin zu Sprach- und Musikausgabe. Der Leser wird dabei durch eine verständliche und humorvolle Sprache ständig bei guter Laune gehalten. Die braucht er auch, denn es kommen auf ihn zahlreiche und vor allem umfangreiche Listings zu. Doch für »Profis« sind die Beispielprogramme oft zu umfangreich. Dagegen ist der über 200 Seiten starke Anhang des Buches wirklich jedem zu empfehlen. Neben Erklärungen und Abhilfen zu allen Fehlermeldungen ist ein umfangreicher Basic-Referenzteil vorhanden. Die Autoren haben die einzelnen Kapitel des Buches gut miteinander verknüpft. Daten aus einer Dateiverwaltung können im darauffolgenden Abschnitt grafisch dargestellt werden und so weiter. Die Verfasser setzen sich dabei erfreulicherweise stark für den IFF-Standard ein.

Der positive Eindruck wird abgerundet durch ein kleines Fachwortlexikon, das dem Anfänger mit Sicherheit einige Fragen beantworten wird.

(Bernhard Carli/rb)

Rügheimer, Spanik, AMIGA BASIC, Data Becker GmbH, 774 Seiten, ISBN 3-89011-209-9, 59 Mark

Diskettendateien im Griff

Mit Hilfe des CLI ist es zum Teil sehr umständlich, Diskettendateien zu kopieren oder zu löschen. Von der Workbench-Ebene aus ist es teilweise sogar unmöglich. CLI-Mate behebt dieses Manko und erleichtert Operationen mit Diskettendateien.

Haben Sie sich auch schon gelegentlich darüber geärgert, daß Sie sich beim Umgang mit dem CLI (Command-Line-Interface) die Finger beim Kopieren oder Löschen von Dateien wundgetippt haben? Oder daß Sie von der Workbench-Ebene aus nicht allzuviel Möglichkeiten haben, um mit den einzelnen Dateien auf Diskette umgehen zu können? Dafür gibt es Gott sei Dank Hilfsmittel. Eines davon ist CLI-Mate, ein Programm, das Ihnen viel Tipparbeit abnimmt und mehr Möglichkeiten als die Workbench-Ebene bietet.

Das Programm dient dazu, Diskettendateien von einer Diskette auf eine andere zu kopieren, Dateien zu löschen oder umzubenennen, Disketten zu formatieren, Directories zu erstellen oder Dateiinhalte zu betrachten. Als kleines Bonbon verfügt CLI-Mate noch über die Option, diese Dateiinhalte auf einem Drucker auszugeben oder IFF-Bilder anzusehen.

Dabei ist das Programm kinderleicht zu bedienen. Alle Funktionen werden mit der Maus angewählt und aktiviert.

Nach dem Starten des Programms und Anlegen einer RAM-Disk, die CLI-Mate selbständig erstellt, befindet sich der Anwender im Hauptmenü (Bild 1), von dem aus sich alle Operationen erledigen lassen. CLI-Mate erkennt automatisch, wie viele Disketten- und Festplattenlaufwerke an den Amiga angeschlossen sind (außer jh0/jh1) und zeigt diese in der Mitte des Bildschirms an. Durch Anklicken des betreffenden Laufwerksymbols bestimmt der Anwender, welches Diskettendirectory geladen werden soll. Dabei kann entschieden werden, ob die betreffende Diskette die Source-(Quell)-Diskette oder die Destination-(Ziel)-Diskette sein soll. Je nachdem, was gewählt wurde, erscheint das Inhaltsverzeichnis in einem Ausschnitt entweder in der linken oder rechten Bildschirmhälfte (Bild 1). Zusätzlich erstellt CLI-Mate auf jeder eingelegten, nicht schreibgeschützten Diskette eine Zusatzdatei mit dem Namen ».fastdir«, in der das Directory der jeweiligen Verzeichnisebene enthalten ist. Sollte das Directory dieser Diskette noch mal von CLI-Mate aus gelesen werden, wird nur diese Datei (sofern vorhanden) geladen und schon ist das Directory auf dem Bildschirm zu sehen. Es muß nicht mehr bei jedem Mal Eintrag für Eintrag gelesen werden, was die Arbeit teilweise wesentlich beschleunigt. Leider ist diese Funktion nicht abschaltbar wie etwa bei »Zing!«. Dies hat den Nachteil, daß in jedem damit bearbeiteten Directory das (je nach Anzahl der Directory-Einträge verschieden lange) Fastdir-File enthalten ist, was natürlich Speicherplatz auf der Diskette frißt.



Bild 1. Hauptmenü des Diskettenhilfsprogramms CLI-Mate

Will man nun einzelne Dateien bearbeiten, müssen diese natürlich erst selektiert (ausgewählt) werden. Dazu fährt man mit dem Mauszeiger auf die betreffende Datei und klickt sie an. Nochmaliges Anklicken deselektiert das File wieder.

Mit der angewählten Datei lassen sich anschließend vielfältige Manipulationen vornehmen. Sie läßt sich löschen (DELETE) oder auf einen anderen Datenträger überspielen (COPY). Das angewählte File kann aber auch von einem Dateiverzeichnis (Directory) in ein anderes verschoben (MOVE) oder umbenannt werden (RENAME).

Das Programm beinhaltet auch eine Funktion, mit der sich Disketten formatieren lassen. Dies ist allerdings nur im ersten und zweiten Laufwerk (df0/df1) möglich. Festplatten oder andere Laufwerke können mit CLI-Mate nicht formatiert werden.



Bild 2. Mit dem Druckermenü lassen sich Art, Form und Gestaltung des Ausdrucks beeinflussen

Wenn Sie gerne den Inhalt einer (Text)Datei auf einen Drucker oder dem Bildschirm ausgeben möchten, hilft CLI-Mate weiter. Dazu muß nur die auszudruckende Datei mit der Maus markiert werden. Ruft der Anwender anschließend die Funktion DISPLAY oder PRINT auf, erscheint (bei Druckerausgaben) ein kleines Auswahlmenü, in dem die Form des Ausdrucks bestimmt werden kann (Bild 2). Dort läßt sich einstellen, ob die Filedaten (Zeit, Datum und Filename) mit ausgegeben werden sollen und an welchen Stellen die Randmarkierungen zu setzen sind. Dieses Menü gestattet eine komfortable Gestaltung des Druckformats. Bildschirmausgaben leitet CLI-Mate natürlich sofort auf den Bildschirm um, ohne ein Eingabemenü zu bringen.

Eine weitere, sehr brauchbare Zusatzfunktion ist der Menüpunkt »SHOWPIC«. Damit können Sie sich Grafikbilder, die im IFF-Format gespeichert sind, von CLI-Mate aus betrachten, ohne extra ein Zeichenprogramm laden zu müssen.

Außerdem kann das Programm, da es über keinen Kopierschutz verfügt, auf jede Diskette kopiert und installiert werden.

CLI-Mate ist ein Programm, welches das Arbeiten mit Diskettendateien erheblich vereinfacht. Es sollte auf keiner Workbench-Diskette fehlen. Die Bedienung des Programms gestaltet sich komfortabel und leicht. Außerdem kann CLI-Mate durch seine Funktionen und die gutdurchdachte Steuerung überzeugen.

Leider kann die Erstellung der Fastdir-Datei nicht abgeschaltet werden, was sich auf den freien Speicherplatz der Diskette auswirkt.

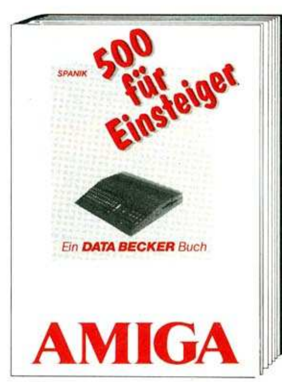
Von diesem kleinen Manko und der leider nur englischsprachigen Anleitung abgesehen kann CLI-Mate guten Gewissens zum Kauf empfohlen werden, nicht zuletzt wegen des angemessenen Preises von etwa 70 Mark. (dm)

Anbieter (Adressen und Preise (zirka 70 Mark) siehe Marktübersicht):
Atlantis, B. Soltau Electronics KG, Instant Replay, Markt&Technik, Softwareland, Soyka Datentechnik, PDC, Jumbo Soft

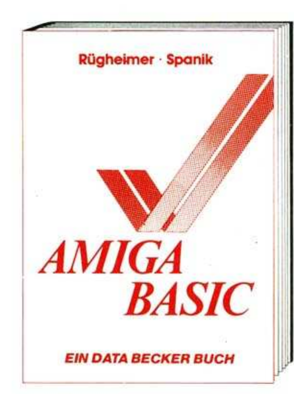
NEUE AMIGA-BUCHHITS



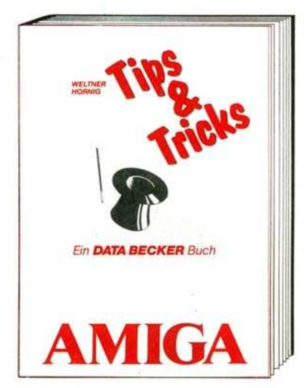
Was leisten die neuen Amigas? Hier finden Sie die Antwort. Unabhängig davon, ob Sie den Amiga schon haben oder den Kauf planen: Dieses Buch bietet Ihnen Entscheidungshilfen, technische Details und jede Menge von dem, was man mit Amiga 500 & 2000 so alles anstellen kann. Eben Informationen, die man braucht, wenn man sich für die neuen Amigas interessiert. Aufbereitet nach einem völlig neuartigen didaktischen Konzept, in einer Sprache, die zum Amiga paßt.
Das können Amiga 500 & 2000
190 Seiten, DM 29,–



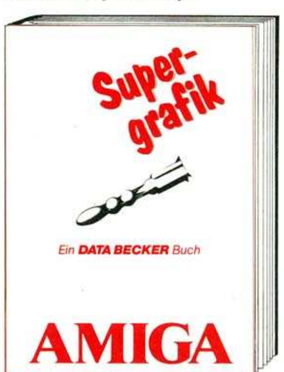
Wählen Sie gleich den richtigen Einstieg zu Ihrem Amiga 500. Denn das Handbuch läßt Sie dabei völlig allein. Versuchen Sie es lieber gleich mit Amiga 500 für Einsteiger. Hier heißt es: Anschließen und loslegen. Verständlich für jedermann zeigt Ihnen dieses Buch: Workbench, Amiga Basic, CLI und AmigaDOS. Locker aufbereitet bietet es Ihnen alles Wissenswerte. Bis hin zu den beim Amiga 500 mitgelieferten Zusatzprogrammen.
Amiga 500 für Einsteiger
343 Seiten, DM 39,–



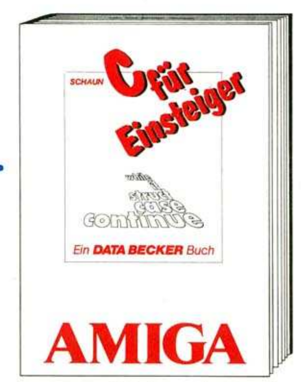
Das erfolgreiche Buch zu AmigaBASIC – jetzt in der Neuauflage! Erweitert um Kickstart 1.2, neuer Workbench und Amiga 500 & 2000. Mit allem, was BASIC-Programmierern Spaß macht: Grafik und Sound, Laden und Speichern von Graficraft-Bildern in BASIC-Programme, sequentielle und relative Dateien, Business-Grafik, Computeranimation, Windows, Umgang mit IFF-Bildern, Sprachausgabe und, und, und. Das Buch für Einsteiger, Aufsteiger und Profis.
AmigaBASIC
Hardcover
774 Seiten, DM 59,–



Amiga Tips & Tricks. Ein Buch, das voller Überraschungen steckt: 64 Farben gleichzeitig auf dem Amiga. Von BASIC aus Zugriff auf die Libraries. Benutzung verschiedener Zeichensätze in BASIC. Sinnvoller Einsatz von Windows, Screens und Menüs. Tips zu einzelnen Grafikbefehlen, Programm- und Amiga-DOS-Routinen! Greifen Sie in die Trickkiste, und schon sind Dinge möglich, die man gar nicht gedacht hätte.
Amiga Tips & Tricks
Hardcover
364 Seiten, DM 49,–



Wer die enorme Grafikfähigkeit seines Amiga ausschöpfen will, braucht entsprechendes Know-how: Grafikprogrammierung mit den vorhandenen BASIC-Befehlen, Nutzung der Libraries, die Register der Grafik-Chips, CAD, Aufbau und Programmierung von Screens, Windows, HAM, Halfbrite und Interlace aus BASIC und C. Informationen, die im Supergrafikbuch zum Amiga mit vielen Programmbeispielen anschaulich vermittelt werden.
Amiga Supergrafik
Hardcover
ca. 700 Seiten, DM 59,–
erscheint ca. 7/87



C an einem Wochenende? Durchaus möglich! Mit C für Einsteiger. Ein Einführungskurs, der Ihnen schnell und einfach die wichtigsten Grundlagen dieser Sprache vermittelt. Vom ersten Programm bis hin zu den Routinen in den Bibliotheken. Mit dem gesamten Sprachumfang und den besonderen Features von C. Zahlreiche Tips & Tricks zur Programmierung und eine Beschreibung der beiden Compiler Lattice C und Aztek runden das Ganze ab.
Amiga C für Einsteiger
Hardcover, 254 Seiten
DM 39,–



Schreiben Sie Ihre Programme in Maschinensprache – und Sie werden sehen, wie schnell ein Amiga sein kann. Das nötige Know-how liefert Ihnen dieses Buch: Grundlagen des 68000, das Amiga-Betriebssystem, Druckeransteuerung, Diskettenoperationen, Sprachausgabe, Windows, Screens, Register, Pull-Down-Menüs ... Und damit Sie auch gleich praktisch arbeiten können, werden die wichtigsten Assembler vorgestellt.
Amiga Maschinensprache
Hardcover, 282 Seiten
DM 49,–

DATA BECKER
Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010

BESTELL-COUPON
Einsenden an: DATA BECKER · Merowingerstr. 30 · 4000 Düsseldorf 1
Bitte senden Sie mir:

zzgl. DM 5,– Versandkosten
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name
Straße
Ort

Durchs Prisma geschaut

AMIGA TEST

Die Zeit des Wartens ist vorbei! Mit Prism liegt jetzt ein Malprogramm für den besonderen H.A.M.-Modus des Amiga vor, das die Bearbeitung dieser speziellen Bilder mit viel Komfort zuläßt und nicht zu teuer ist.

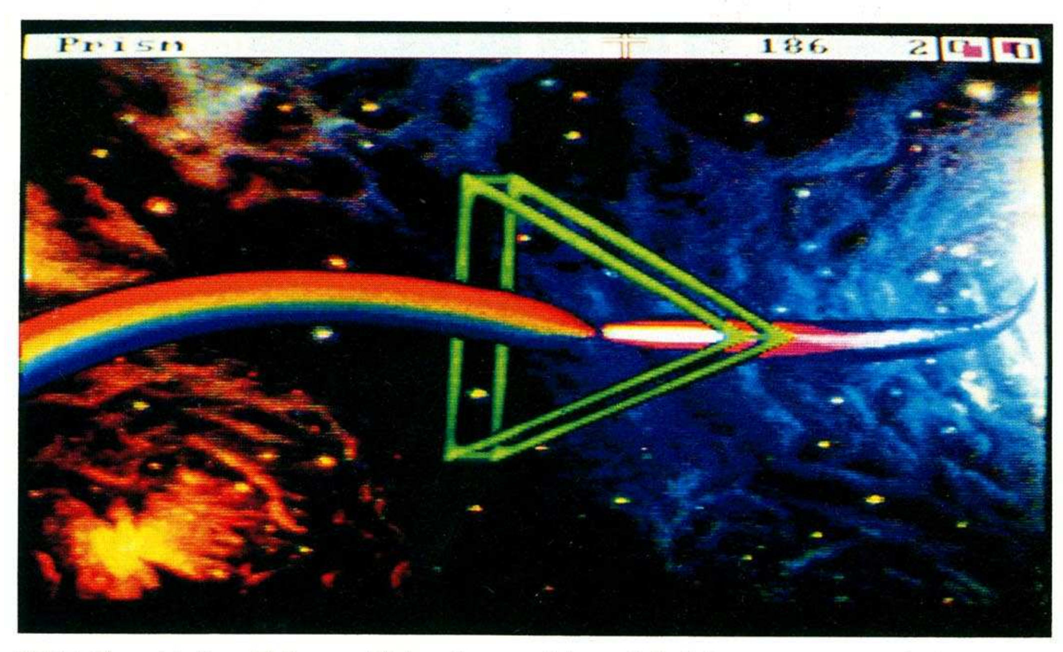


Bild 1. Verschiedene Teile von Bildern lassen sich nach Belieben zusammenmischen

Eine einzigartige Eigenschaft des Amiga in bezug auf die Darstellung von Grafiken auf dem Bildschirm ist der Hold-and-Modify-Modus (H.A.M.). Kein anderer Computer verfügt über eine ähnliche Einrichtung. Hardwaremäßig verfügt der Amiga über 32 Farbregister, von denen jedes 12 Bit breit ist. Die Einstellung der Farbintensität der drei Grundfarben Rot, Grün und Blau, die dazu benutzt wird, um einem jeden Punkt am Bildschirm die Farbe zu geben, geschieht über diese 12 Bit. Wobei jeweils vier Bit für eine der Grundfarben benötigt wird. Es bestehen demnach 16 verschiedene Möglichkeiten, diese vier Bit für jede Grundfarbe zu setzen (16 verschiedene Abstufungen einer Grundfarbe sind verfügbar). Insgesamt lassen sich somit 16 x 16 x 16 = 4096 Farben auf dem Amiga darstellen. Aufgrund der Hardware-Eigenschaften des Grafik-Chips Denise sind diese 4096 Farben gleichzeitig nur in einem Modus darstellbar. Das ist der H.A.M.-Modus, den es in zwei verschiedenen Auflösungsstufen gibt: 320 x 200 Punkte und 320 x 400 Punkte. Dabei ist letzteres der sogenannte H.A.M.-Interlaced-Modus, da der Amiga außerdem 400 (beziehungsweise 512 Zeilen in der PAL-Version) Zeilen am Bildschirm nur durch das Umschalten zweier Halbbilder erzeugen kann. Dabei entsteht das berühmt berüchtigte Flimmern. Um nun die Arbeit mit dem Malprogramm Prism verstehen zu können, muß man diese Grundlagen stets im Hinterkopf behalten, denn daraus resultieren einige besondere Eigenheiten des Programms, die man sonst falsch verstehen würde. Eine sehr auffällige Konsequenz des vorher erwähnten ist, daß im H.A.M-Modus die Farbwahl zwischen zwei Pixeln (kleinste Bildpunkte), die rechts beziehungsweise links nebeneinander liegen in gewisser Weise eingeschränkt ist. Wenn man einen Punkt in einer bestimmten Farbe setzt, so kann man den nächsten direkt danebenliegenden Punkt nicht in einer völlig anderen Farbe setzen. Das sieht man ganz einfach, wenn nach dem Starten von Prism versucht wird, auf dem leeren schwarzen Bildschirm einen weißen Punkt zu setzen. Links vom weißen Punkt ist der normale schwarze Hintergrund zu sehen. Rechts jedoch hat der Amiga einen Übergangspunkt in Blau gefärbt. Das Bild verschwimmt also je nach Farbwahl mehr oder weniger stark.

Grundsätzlich liegt es also nicht an Prism, sondern am H.A.M.-Modus selbst, wenn man ziemlich schnell feststellt, daß das Programm für reines Freihandzeichnen relativ ungeeignet ist. Es gibt natürlich einen entsprechenden Menüpunkt zum Setzen von Punkten und Zeichnen von Linien in den verschiedensten Pinselstärken, er erlangt jedoch hier nie eine so große Bedeutung wie etwa bei Deluxe-Paint. Der H.A.M-Modus ist ein hervorragender Weichzeichner von Konturen. Eine ganz wichtige Bedeutung kommt dem Programm damit durch die Nachbearbeitung von digitalisierten Bildern zu (Bild 1). H.A.M.-Bilder sind zwar nicht durch ihre Auflösung (höchstens 320 Spalten) wirklichkeitsnah, sondern durch die Menge an Farben, die subjektiv im Auge des Betrachters ein Bild mit höherer Auflösung, aber weniger Farben, unechter erscheinen läßt. Dies ist eine erstaunliche und in der Psychologie begründete, jedoch für den H.A.M.-Modus optimale Tatsache.

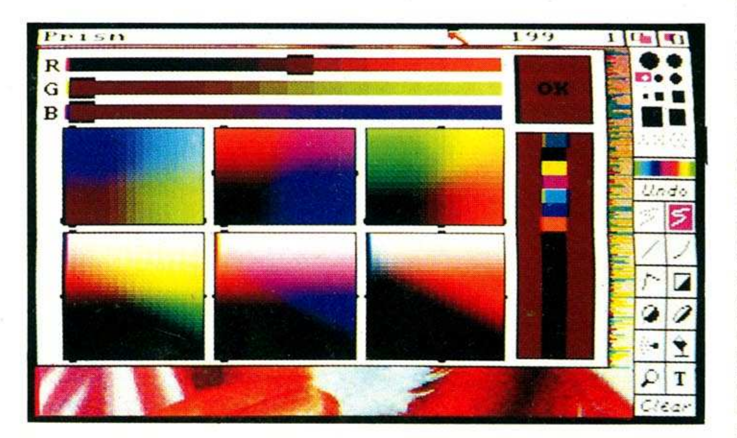


Bild 2. Die Farbpalette von Prism zeigt, was im Amiga steckt und ist für den H.A.M.-Modus optimal aufgebaut

Prism läuft mit mindestens 512 KByte Speicher und einem Floppy-Laufwerk. Weitere Laufwerke erhöhen natürlich den Komfort und zusätzlicher Speicherplatz beschleunigt das Arbeiten mit dem Programm. Da Prism nicht kopiergeschützt ist, kann man es auch auf Festplattenlaufwerke installieren und eigene Sicherheitskopien erstellen. Eine Methode des Herstellers, die Nachahmer finden sollte.

Sehr auffällig zeigt sich das Programm, wenn die Farbpalette aufgerufen wird (siehe Bild 2). Hier kann das Programm und der Amiga einmal so richtig zeigen, was in ihnen steckt. Die Palette besteht im wesentlichen aus vier Teilen.

Prism überall installieren

Die sechs großen Farbfelder sind zum Aussuchen der Farben optimal aufgebaut: man sieht Farbverläufe und ähnliche oder zueinander passende Farben und kann diese direkt anklicken. Für das feinere Verstellen einer Farbe dienen die drei RGB-Schieberegler im oberen Teil der Palette. An der rechten Seite liegt der »Last Color«-Bereich. Die momentan gewählte Zeichenfarbe wird in dem großen umgebenden Bereich angezeigt, während der zuerst schwarze Balken in der Mitte ein »First in first out«-Speicher für die letzten 15 benutzten Farben ist. So kann man sehr einfach wieder auf eine schon einmal gebrauchte Farbe zurückgreifen.

Die Lupenfunktion wird einige Anwender auf den ersten Blick enttäuschen. Man sollte nicht Deluxe-Standard anwenden, bei Prism gibt es nur eine Vergrößerungsstufe. Mehr braucht man allerdings auch gar nicht. Auf pixelgenaues Zeichnen kommt es dabei nicht an; das Überblenden von einer Farbe in eine andere ist gefragt (Farbübergang). Das einzige was voll von Deluxe-Paint übernommen wurde, ist zusätzlich zur Bedienung über die Maus noch die Belegung der Tastatur mit Funktionsaufrufen. Bei den Funktionstasten F1 bis F10 braucht man sogar fast überhaupt keine Umgewöhnungszeit. Richtig ausreizen läßt sich Prism jedoch erst, wenn man über alle Befehle auf der Tastatur Bescheid weiß. Viele grundlegende Optionen sind nur über die Tastatur zu erreichen. Dies soll keine Schikane für ungeübte Benutzer sein, sondern ist der einzige Quasi-Kopierschutz für das Programm. Da man ohne Handbuch sonst wenig damit anfangen kann. Sehr komfortabel zeigt sich Prism dagegen, wenn es darum geht, verschiedene Bilder oder Teile von Bildern miteinander zu mischen (Bild 3). Herauskopieren, Übereinanderlagern oder Bereiche und Farben festhalten ist mit dem Programm kein Problem. Die »Undo«-Funktion ist äußerst gut durchdacht: Neben der normalen Funktion wie in Deluxe-Paint gibt es auch Undo-Puffer für Brushes oder kopierte Bildbereiche.



Bild 3. Digitalisierte Bilder kann man mit Prism auf vielfältige Weise nachbearbeiten

Prism bietet außer der normalen Druckoption noch einen Druck des gesamten Bildes in jeweils einer Grundfarbe. Das Bild kann also nur in Grün, Blau oder Rot auf den Bildschirm gebracht und dann gedruckt werden. Besitzt man einen Farbdrucker und verwendet durchsichtiges (pergamentenes) Papier, so kann man die drei Ausdrucke später übereinanderlegen und erhält so ein Farbbild.

Einmal anders gedruckt

Prism ist zur Zeit das einzige Malprogramm, welches die Verwendung von beliebig vielen Fonts (Schriftarten) unterstützt und das in bis zu 16 Farben. Deluxe-Paint stellt zwar auch mehrere Fonts zur Verfügung, diese werden aber über ein Pull-Down-Menü angesprochen. Man kann zugegebenermaßen somit eine ganze Menge Fonts verwalten, wenn das Menü allerdings voll ist, können keine weiteren Fonts mehr aufgenommen werden. Prism verfügt aus diesem Grund für das Nachladen und Aussuchen der Schriften über ein Auswahlfenster wie für das Laden der Gesamtbilder auch (mit Rollbalken etc.). Außerdem können durch den 4096-Farben-Modus selbsterstellte Zeichensätze vom »Calligrapher« übernommen werden. Diese dürfen ein Muster (Pattern) im Hintergrund haben, das bis zu 16 Farben verwendet.

Für alle, die einmal wissen möchten, wie H.A.M.-Bilder im IFF-Standard aufgebaut sind oder mit welcher Speicheraufteilung das Programm intern arbeitet, sind diese Informationen im Handbuch zu finden. Prism kann nebenbei die H.A.M.-Bilder auf zwei verschiedene Arten auf Diskette speichern. Einmal im normalen IFF-Standard, der dann auch von anderen Programmen verstanden wird; gesetzt den Fall, das Programm arbeitet überhaupt im H.A.M.-Modus. Zweitens erlaubt Prism ein spezielles Speicherformat mit 50 bis 80 Prozent Platzersparnis auf Diskette.

Ansonsten ist das Programm zwar etwas gewöhnungsbedürftig in der Bedienung (teilweise über Tasten), aber wenn man sich damit erst einmal eingearbeitet hat, kommt man schnell zu den Ergebnissen, die man erzielen möchte. Für die Bearbeitung von H.A.M.-Bildern auf dem Amiga ist Prism zur Zeit ein »Muß« für grafikbegeisterte Computerbesitzer. (jk)

Anbieter (Adressen siehe Marktübersicht): Markt & Technik, Softwareland, Jumbo Soft
Preis: rund 159 Mark

Assembler im Wettbewerb

AMIGA TEST

Trotz des steilen Aufstiegs der Sprache »C« spielt die Assemblerprogrammierung auf dem Amiga keineswegs nur eine Nebenrolle. Für schnelle und kompakte Programme ist sie noch die beste Alternative. Was bietet der Markt nun denjenigen Programmierern, die mit Bits und Bytes experimentieren wollen?

Mit einem Assembler kann man alle Möglichkeiten nutzen, die ein Computer zu bieten hat, muß sich dabei aber auf ein recht niedriges Niveau der Programmierung begeben. Selbst einfache Dinge — wie das Ausgeben kurzer Texte auf dem Bildschirm — erfordern in Assembler-Sprache oft größere Gedankenarbeit. Logische Fehler im Programm sind sehr leicht einzubauen und führen meist zu einem Systemabsturz. Es ist deshalb demjenigen, der bisher weder Assemblererfahrung noch Kenntnisse der Computerhardware hat, nicht unbedingt zu raten, diesen Einstieg ausgerechnet mit dem Amiga zu wagen. Er kommt mit »C« wahrscheinlich eher an sein Ziel.

Dem Umsteiger, der schon dem 6502 oder Z80 Beine gemacht hat, bieten sich allerdings mit dem 68000-Mikroprozessor und der Hardware des Amiga Möglichkeiten, von denen er bisher nur zu träumen wagte.

Für den Test standen uns der »Devpak Assembler« von Hisoft, Metacomcos »Makro-Assembler« und der Amiga-Veteran »Seka« von Kuma zur Verfügung. Ein wesentliches Kriterium bei der Beurteilung eines Assemblers sollte die Zeit sein, die der Programmierer von der Idee bis zum ablauffähigen Programm braucht. Außer der Übersetzungsgeschwindigkeit spielt hier auch der Aufbau des Handbuchs eine nicht geringe Rolle. Während Kuma hier eine hart an der Grenze zum Stichwortverzeichnis liegende Beschreibung liefert, kann man die Dokumentation der beiden anderen Assembler durchaus als ausführlich bezeichnen.

Das Handbuch von Hisoft beginnt mit einer schnellen Einführung in die Bedienung des Devpak-Assemblers am Beispiel der Übersetzung eines Demoprogrammes. Programmierern, die prinzipiell erst das Handbuch zur Hand nehmen, wenn sie nicht mehr weiterkommen, ist damit erst mal gedient. Neben der Beschreibung von Editor, Assembler und Debugger findet man in diesem Handbuch außerdem eine Einführung in die Arbeitsweise von AmigaDos und CLI. Es ist in einfacher, englischer Sprache geschrieben und macht den Leser anhand ausführlicher Beipiele schnell mit den Funktionen des Assemblers vertraut. Metacomcos Dokumentation, ebenfalls in Englisch, gibt sich da schon etwas bescheidener. Nicht unmittelbar einsichtige Ausführungen und fehlende Beispiele erschweren etwas das Verständnis der Besonderheiten bei der Arbeit mit diesem Prozessor. Dies ist zwar nicht unbedingt Aufgabe des Handbuches, erleichtert den Einstieg jedoch beträchtlich. Besitzer des Seka-Assemblers ohne viel Erfahrung mit solchen Assemblern werden schließlich einige Zeit mit anderen Publikationen verbringen, um herauszufinden, was mit den im »Handheft« beschriebenen Funktionen möglich ist.

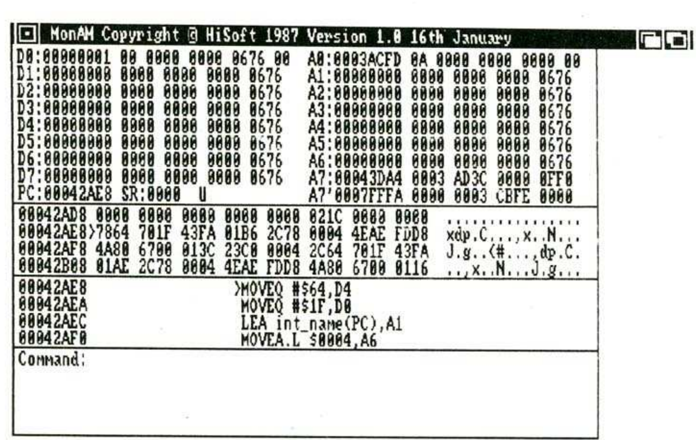


Bild 1. Fehlersuche mit dem Hisoft-Debugger

Übersetzung in drei Phasen

Wie bei den meisten höheren Programmiersprachen entsteht beim Makro-Assembler ein lauffähiges Programm in drei Phasen.

Die erste Phase ist das Erstellen (Editieren) des Quelltextes. Metacomco liefert dazu den Texteditor »ed«, der sich auch schon auf der Workbench befindet. Programmierer, die sich schon an diesen Bildschirmeditor gewöhnt haben, brauchen sich also nicht umzustellen. Die zweite Phase ist der eigentliche Assembliervorgang. Das Quellprogramm muß sich hierfür auf der Diskette (auch Ram-Disk) befinden. Beim Aufruf des Assemblers angegebene Parameter beeinflussen den Übersetzungsvorgang. In der dritten Phase erzeugt der Linker aus dem vom Assembler erzeugten Objektprogramm ausführbaren Maschinencode. Die Hauptaufgabe des Linkers ist jedoch das Verketten (Linken) von Objektmodulen. Sie können sich so ein Modul mit oft benötigten Routinen programmieren, dieses assemblieren und erst durch den Linker mit dem aktuellen Programm verketten. Metacomco liefert als Linker den langsamen Alink. Der als Public-Domain-Software erhältliche und schnellere Blink wäre hier besser geeignet.

Da beim Aufruf von Assembler und Linker eine Reihe von Angaben möglich sind, können die Anweisungen ziemlich lang werden. Um nicht für jeden Aufruf alles neu einzugeben, empfiehlt sich die Erstellung einer Batch-Datei. Eine solche Datei enthält dann die Aufrufe von Assembler und Linker einschließlich Parameter. Mit dem Befehl »execute« wird dann der gesamte Übersetzungsvorgang gestartet. Auf der von Metacomco mitgelieferten Diskette befindet sich bereits die Batch-Datei »make« für das Assemblieren und Linken eines Demoprogramms. Für die Übersetzung eines anderen Programms muß »make« durch mehrmaliges Ersetzen der Programmnamen geändert werden. Das ist nicht sehr komfortabel und kann verbessert werden.

Komfortable Fehlersuche

Nicht jedes Programm muß mit anderen Objektmodulen verkettet werden. Wenn der Assembler gleich ausführbaren Maschinencode erzeugen würde, wäre das zeitaufwendige Linken nicht notwendig. Die Programmierer von Hisoft haben das wohl eingesehen und deshalb kann der Devpak-Assembler sowohl Linkmodule als auch ausführbare Programme erzeugen. Ein weiterer Unterschied zum Konkurrenten von Metacomco ist die Integration von Editor und Assembler. Letzterer kann über einen Menüpunkt direkt vom Editor aufgerufen werden. Da sich beim Amiga mehrere Programme gleichzeitig im Speicher befinden können, liegt der Vorteil der Integration weniger in der leichteren Bedienung als an der Ersparnis vom Speicherplatz.

Devpak ist ein echtes Amigaprogramm. Es nutzt als einziger der Testkandidaten den Komfort aus, den die Benutzeroberfläche Intuition bietet. Über Menüfunktionen oder Tastenkombinationen hat man Zugriff auf einige Editorfunktionen, kann Textmodule laden, speichern, einfügen oder auch Teile suchen und ersetzen, oder den Assembler starten. Für das Verketten der Programmodule stehen schließlich zwei Linker, Alink und Blink, zur Verfügung.

Anders als bei den höheren Programmiersprachen, wo ein großer Teil der Programmfeh-

Wir machen Preise!

Sie zahlen: keine Einfuhrumsatzsteuer, keine Verpackung, keine Versandspesen

Artikel	DM
APACUS BOOKS	
AMIGA BASIC IN&OUT BOOK	59
AMIGA TRICKS&TIPS BOOK	49
ABSOFT	
AC BASIC Compiler	419
AC FORTRAN Compiler	639
ACCESS ASSOCIATES	
ALEGRA 2MB RAM extern	1198
ALEGRA 512KB RAM	869
ACCESS SOFTWARE	
GOLF Datendisk #1	49
LEADER BOARD (Golf Game)	79
TENTH FRAME	79
ACCOLADE	
MEAN 18 Golf	59
ACTIVISION	
BORROWED TIME game	69
CHAMPIONSHIP BASEBALL	69
CHAMPIONSHIP BASKETBALL	69
CHAMPIONSHIP FOOTBALL	79
CHAMPIONSHIP GOLF	79
HACKER I game	55
HACKER II game	85
LITTLE COMPUTER PEOPLE	65
MINDSHADOW	69
PORTAL	79
SHANGHAI	69
TASS TIMES IN TONETOWN	98
THE MUSIC STUDIO	98
AEGIS DEVELOPMENT	
AEGIS Animator & Images	269
AEGIS Artpack-1	59
AEGIS DIGA!	179
AEGIS Draw	239
AEGIS DRAW PLUS	489
AEGIS IMAGES	79
AEGIS IMPACT	289
AEGIS SONIX	179
AROZOK'S TOMB adv.	109
DEMO Animator	15
DEMO Draw	15
VIDEOSCAPE 3-d	?
AMAZING COMPUTING	
AMAZING COMPUTING je	10
AMI PROJECT	
AMI PROJECT MAGAZINE je	10
AMIGALAND	
PUBLIC DOMAIN Disks je	10
AMISOFT	
DISKZINE Nr. AA-AG	25
ANAKIN RESEARCH	
DEMO EASYL Grafiken	10
EASYL Zeichentablett	979
ANALOG PRECISION	
MEGAAMIGA 2MB RAM	1395
ANCHOR AUTOMATION	
COMPUTER MAILBOX	695
LIGHTNING 24 Modem	1095
OMEGA 80 Amiga Modem	495
SIGNALMAN Express Modem	995
APPLIED VISIONS	
FUTURE SOUND II	389
ARBORSOFT	
2+2 HOME MANAGEMENT	198
ARRAY'S INC.	
MASTER AM DOS Buch	39
ARTWORX SOFTWARE	
BRIDGE 5.0	89
STRIP POKER	76
ASDG	
FLOPPY ACCELERATOR	89
ASSOCIATED COMPUTER	
BRUSH WORKS (3 disks)	69
GRADE MANAGER	198
MUSIC STUDENT	128
QUIZ MASTER	179
BANTAM	
DOS EXPRESS	59
DOS Manual Book	59
USER GUIDE graphic/sound	49
BAUDVILLE	
VIDEO VEGAS	59
BETHESDA SOFTWORKS	
DEMO Gridirion	15
GRIDIRON	129
BYTE BY BYTE	
FINANCIAL PLUS	599
INFOMINDER	229
PAL 20Meg. Expansion	6495
PAL 43Meg. Expansion	7995
PAL JR 20Meg+1MB RAM	3995
PAL Standard	3995
PROTOTYPE CARD	179
TIC Clock (AMIGA Uhr)	149
C-LTD.	
20 Meg. Hard Drive SCSI	1995
50 Meg. Hard Drive SCSI	3495
aMega Board 1MB Fast RAM	795
Harddiskcontroller SCSI	998
TIME SAVER	178
CAPILANO COMPUTER SYSTEMS	
LOGIC WORKS (Logic CAD)	198
CENTRAL COAST SOFTWARE	
DISK TO DISK	109
DOS to DOS	118
CLASSIC IMAGE	
DIABLO	59
COMMODORE	
AMIGA Jahrbuch 1987	10

Artikel	DM
AMIGATERM	109
COMMODORE COMPUTING INT.	
BUSINESS AMIGA MAGAZINE	6
COMPUTING MAGAZINE je	6
COMPLETE DATA AUTOMATION	
SPELLING CHECKER	149
COMPUMED	
MIRROR HACKER PACKAGE	109
MIRROR Kopierprogramm	109
COMPUTE BOOKS	
ADVANCED AMIGA BASIC	49
AMIGA 1st COMPUTER	49
AMIGA APPLICATIONS	49
AMIGA C FOR BEGINNERS	49
AMIGA PROGRAMMERS GUIDE	49
BEGINNERS GUIDE AMIGA	49
C PROG. GRAPHICS BOOK	59
ELEMENTARY AMIGA BASIC	49
FLIGHT SIM. ADV. BOOK	49
INSIDE AMIGA GRAPHICS	49
KIDS AND THE AMIGA	49
USING AMIGA DOS	49
COMSPEC	
COMSPEC 1 MByte RAM	898
COMSPEC 2 MByte RAM	1198
COPPERSTATE	
QUICK NIBBLE Duplicator	89
COSMI	
SUPER HUEY	68
CREATIVE MICRO SYSTEMS	
KICK START ELIMINATOR	279
CRYSTAL ROSE	
ANALYTIC ART	139
DARK HORSE	
CHESSMATE Schach	59
DELTA RESEARCH	
FORTH	198
DIAMOND SOFTWARE	
C MONITOR V 200	189
DEMO Investor	15
INVESTOR	1599
DIGITAL CREATIONS	
D'BUDDY	169
DIGITAL LINK	149
GIZMOS 2.0	149
DIGITAL SOLUTIONS	
LPD FILER	279
LPD PLANNER	279
LPD WRITER	279
DISCOVERY SOFTWARE	
DISCOVERY-SHELL	119
DISKWORKS	199
DX-16	119
GRABBIT	59
KEY-GENIE	119
MARAUDER II	79
DTM	
AMIGA1000 Druckerkabel	39
DEMO DiskCovery	15
DisCovery Disk Editor	198
GRAFIKBILDERDISK #	15
EAGLE TREE SOFTWARE	
GRAFICS-CONVERTER+EDITOR	109
ECE RESEARCH	
ECE Midi Interface+RS232	129
REFERENCE PAK (5 Cards)	39
ELECTRONIC ARTS	
ADVENTURE CONSTRUCTION	69
ARCHON game	89
ARCTICFOX game	59
AUTO DUEL sim.	128
BARDS TALE adv.	89
BLACK CAULDRONS game	128
CHESSMASTER 2000 Schach	79
DELUXE MUSIC	239
DELUXE PAINT II	239
DELUXE PRINT & Data #1	149
DELUXE VIDEO II	239
DEMO Adventure	15
DEMO Bards Tale	15
DONALD DUCKS PLAYGROUND	79
EARL WAVER BASEBALL	109
INSTANT MUSIC	98
INSTANT MUSIC DATA #1	89
MARBLE MADNESS	59
RETURN TO ATLANTIS	109
EPYX	
ROGUE Adventure	68
TEMPLE OF ASPHAI	68
WINTERGAMES	68
WORLD GAMES	68
EQUAL PLUS	
FINANCIAL PLUS	598
FINALLY SOFTWARE	
DR. XES	119
NANCY Spelling checker	119
PHASAR Finanz management	129
SENOR TUTOR	119
TALKER wordprocessor	149
FIREBIRD	
GUILD OF THIEVES Adv.	99
KNIGHT ORC adv.	99
STAR GLIDER game	99
THE PAWN Adventure	99
JEWELS OF DARKNESS	79
FIRST BYTE	
FIRST SHAPES AMIGA	
MAD LIBS AMIGA	128
MATH TALK AMIGA	128
SPELLER BEE AMIGA	128
TALKING NOTEBOOK	128

Artikel	DM
GOLD DISK INC.	
GOLD SPELL	89
PAGE-SETTER (Umlaute)	298
PAGESETTER LASERSCRIPT	89
GOLDEN HAWK	
GOLDMIDI Interface	169
GRAFOX OF ENGLAND	
LOGISTIX	298
HASH ENTERPRISES	
'DISNEY' 3D-Animator	798
IDEAS CREATED	
JITTER-RID Filterglas	29
IMPULSE	
PRISM	149
PROCLIP	59
INFINITY SOFTWARE	
GALILEO Planetarium	200
GRAND SLAM TENNIS	69
INFOCOM	
BALLYHOO mystery	89
BUREACRACY comedy	89
ENCHANTER adv.	69
HITCHHIKERS GUIDE Adv.	89
MIND FOREVER	89
MOONMIST	89
TRINITY	89
WISHBRINGER	49
ZORK I Fantasy Adv.	89
ZORK II Fantasy Adv.	98
ZORK III Fantasy Adv.	98
ZORK Trilogy (3 in 1)	149
INNOVATRONICS INC.	
DEMO POWER WINDOW	10
POWER WINDOWS	189
INTELLECTUAL SOFTWARE	
AMERICAN HISTORY ADV.	119
FRENCH GRAMMAR	99
HOW A BILL BECOMES A LAW	119
SPANISH GRAMMAR I	99
SPANISH GRAMMAR II	99
SPANISH GRAMMAR III	99
STARTING A NEW BUSINESS	119
US GEOGRAPHIE ADVENTURE	129
WORLD GEO ADV. AFRICA	129
WORLD GEO ADV. AMERICA	129
WORLD GEO ADV. ASIA	129
WORLD GEO ADV. EUROPA	129
WORLD HISTORY ADVENTURE	129
INTERACTIVE ANALYTIC NODE	
EXPERT SYSTEM KIT	149
THE EXPLORER Debugger	98
INTERACTIVE SOFTWORKS	
CALLIGRAPHER	209
ISM INC.	
SURGEON educational	98
JAGHWARE INC.	
ALIEN FIRES adventure	86
JUMPDISK	
Diskmagazin	25
KENT ENGINEERING	
MACROMODEM software	149
LATTICE	
dBC III	289
LATTICE C Compiler	448
MAC LIBRARY	189
UNICALC	169
MANX SOFTWARE	
AZTEC C Commercial	1098
AZTEC C Developers	698
AZTEC C Personal	439
MERIDIAN SOFTWARE	
ZING KEYS	109
ZING! V1.2	169
METACOMCO	
METACOMCO Lisp	398
METACOMCO Makro Assembler	169
METACOMCO Pascal	228
METACOMCO shell	109
METACOMCO Toolkit	89
MICRO ILLUSIONS	
DEMO Dynamic CAD	15
DISCOVERY	109
DISCOVERY MATH educ.	98
DISCOVERY TRIVIA game	98
DYNAMIC CAD	998
DYNAMIC WORD	398
FAIRY TALE ADVENTURE	98
FIRE POWER game	59
GALACTIC INVASION game	59
LAND OF LEGENDS adv.	109
TURBO car racing	59
MICRO SYSTEMS SOFTWARE	
ANALYZE 2.0 spreadsheet	339
ANALYZE spreadsheet	198
BBS Bulletin Board	219
FLIPSIDE printer driver	129
ONLINE Telekommunikation	149
ORGANIZE database	219
SCRIBBLE	219
MICROBOTICS	
MULTIFUNCTION MODULE	249
STARBOARD 2 MByte	1595
MICROPROSE	
GUNSHIP game	86
MICROSEARCH	
CITY DESK Desktop P.	298
MICROSMITHS	
FAST FONTS	79
TxED Editor	84
MICROSOFT PRESS	
THE AMIGA BOOK	59
MILES COMPUTING	

Artikel	DM
QUINTETTE strategy game	89
MIMETICS	
MIDI INTERFACE	109
PRO STUDIO	339
SOUND SCAPE Sampler	219
MINDSCAPE	
BALANCE OF POWER	79
DEFENDERS OF THE CROWN	79
DEJA VUE	79
HALLEY PROJECT	79
KING OF CHICAGO game	89
S.D.I. game	79
SHADOWGATE	79
SINBAD game	79
UNINVITED	79
NEW HORIZONS	
DEMO PRO WRITE & FLOW	198
PRO WRITE	198
NEW-TEK	
DEMO DIGI-VIEW	15
DIGI PAINT	119
DIGI-VIEW Digitizer	449
NORTHEAST SOFTWARE	
ORDER Desktop organizer	109
PUBLISHER 1000	449
ORIGIN SYSTEMS	
OGRE	69
ULTIMA III	129
ULTIMA IV	129
OTHER GUYS	
GREAT STATES educ.	79
OMEGA FILE database	169
PROMISE spellcheck	109
OXXI INC.	
ENCORE mouse/keyb/macro	89
MAXICOM communication	98
MAXIDESK DTP calendar	128
MAXIPLAN Lotus komp.	398
MAXIPLAN PLUS	498
WOW disk cache system	89
PAR HOME	
DIMMER screen dimmer	39
DISK PRO PLUS	69
EXPRESS PAINT	198
OUTLINE processor	98
PARTY SOUND INC.	
60's GREATEST (45 min.)	49
70's GREATEST (45 min.)	49
80's GREATEST (50 min.)	49
BARBERSHOP (65 min.)	49
BEATLES Part 1 (40 min.)	49
BEATLES Part 2 (40 min.)	49
BEETHOVEN (40 min.)	49
BILLY JOEL (65 min.)	49
BROADWAY THEME Y(40 min.)	49
CHRISTMAS (50 min.)	49
CHURCH MUSIC (50 min.)	49
CLASSICAL #1 (40 min.)	49
CLASSICAL #2 (40 min.)	49
CLASSICAL #3 (35 min.)	49
CLASSICAL #4 (30 min.)	49
CLASSICAL #5 (50 min.)	49
COUNTRY #1 (45 min.)	49
COUNTRY #2 (50 min.)	49
GOLD&PLATINUM (60 min.)	49
KENNY RODGERS (45 min.)	49
MOVIE THEMES (40 min.)	49
NOSTALGIA (45 min.)	49
POLKA PARTY (40 min.)	49
R.RODGERS SONGBOOK	49
ROCK Part 1 (50 min.)	49
ROCK Part 2 (50 min.)	49
SYMPHONY JUKEBOX	49
SYMPHONY MUSIC VIDEO	49
TV THEMES (35 min.)	49
PECAN SOFTWARE	
BASIC P-Code Basic	198
BASIC Prof. pak	398
FORTRAN 77	198
FORTRAN Prof. pak	398
MODULA-2	198
MODULA-2 Prof. pak	398
PASCAL Prof. pak	398
UCSD Pascal	198
PILOT	
MOUSE HIDE Leder Pad	29
PPI	
J-1000 SUPERJOYSTICK	19
PRECISION SOFTWARE	
SUPERBASE Personal	198
PROGRESSIVE PERIPHERALS	
CLI MATE Diskutility	69
MEGABOARD II 2 MByte RAM	1198
PSYGNOSIS	
BARBARIAN	69
BRATACASS + ARENA	79
TERRORPODS	69
S.S.I.	
COMPUTER BASEBALL	68
KAMPFGRUPPE strategy	122
PHANTASIE Adventure	98
ROADWAR 2000 game	98
SAMS BOOKS	
ADVANCED C PRIMER	69
C PRIMER PLUS	69
INSIDE THE AMIGA	69
SEDONA SOFTWARE	
MONEY MENTOR	249
SIDE EFFECTS INC.	

Artikel	DM
SIDE ARM (2 MByte+20 Meg)	3995
SIERRA ONLINE	
KINGS QUEST 1	89
KINGS QUEST 2	89
KINGS QUEST 3	89
SLIPPED DISK	
BASIC KEYBOARD OVERLAY	19
DOS KEYBOARD OVERLAY	19
SOFTWARE FACTORY	
BLACK JACK game	69
POKER	79
SOLITAIRE	49
SOFTWOOD COMPANY	
SOFTWOOD FILE Database	198
SOFTWOOD FILE SG	289
SOFTWOOD WRITER	198
SOUND LIBRARIES INC.	
COMPOSER DISK Volume I	69
COMPOSER DISK Volume II	69
COMPOSER DISK Volume III	69
INSTR. DISK B-3 Organ	69
INSTR. DISK Melotron	69
INSTR. DISK Steeldrums	69
SOUND DISKS 1-49 je	49
WAVEBUILDER Supersound	69
SPEECH SYSTEMS	
DIGITIZED SONGS Disks je	59
SUBLOGIC	
FLUGSIMULATOR II	109
JET simulator	109
Scenery Disk #7	49
SNENERY DISKS (1-6) Paket	199
SUNRIZE INC.	
DEMO Perfect Sound	15
PERFECT SOUND	169
STUDIO MAGIC	98
SUPRA COORPERATION	
SUPRA DRIVE 20 MEG	1998
SUPRA DRIVE 30 MEG	2298
SUPRA DRIVE 60 MEG	3698
TAURUS IMPEX	
ACQUISITION	639
DEMO Acquisition	15
TDI SOFTWARE	
AMIGA Editor	109
DEMO Modula-2	15
EXAMPLES Disk	49
KERMIT SOURCE	58
MODULA-2 Commercial	689
MODULA-2 Developer	339
MODULA-2 Standard	198
THE GRID	109
TEAK TECH DESIGNS	
MD-120 3½ Teakdiskbox	79
MD-60 3½ Teakdiskbox	59
MD-60L 3½ Teakdiskbox	79
TRUE BASIC	
Compiler Runtime	339
Developers Toolkit	109
LIB 3D Graphics	109
LIB Advanced Strings	109
LIB Sorting & Searching	109
PRG Algebra 1	109
PRG Algebra 2	109
PRG Calculus	109
PRG Discrete Maths	109
PRG Pre Calculus	109
PRG Probability Theorie	109
PRG Trigonometrie	109
PRG True Stat	169
TRUE BASIC language	198
UNICORN	
AESOP'S FABLES educational	98
ANIMAL KINGDOM educ.	98
DECIMAL DUNGEON educ.	98
FRACTION ACTION educ.	98
KINDERAMA EDUCATIONAL	98
MATH WIZARD educ.	98
UNISON WORLD	
ART GALLERY #1	59
ART GALLERY #2	59
PRINTMASTER PLUS	109
VERSASOFT	
dBMan dBase III komp.	339
DEMO dBMan	15
VIP TECHNOLOGIES	
VIP PROFESSIONAL	319
VIZA SOFTWARE	
DEMO VIZAWRITE Desktop	15
VIZAWRITE Desktop dtsch.	198
VIZAWRITE DESKTOP engl.	198
WAVE PAD	
MOUSE PAD (9½" x 11")	18
WAVETABLE	
SOUND LIBRARIES je Disk	69
WESTCOM INDUSTRIES	
DISK FILE ORGANIZER	109
HARDDISK BACKUP	149
XEROX	
XEROX 4020 Color Ink Jet	3495
ZIRKONICS	
TEXT ENGINE	229
ZUMA CROUP	
TV TEXT Titelanimation	228
TV-TEXT 3-D Textanim.	228
ZUMA FONTS Vol. 1	89
ZUMA FONTS Vol. 2	89
ZUMA FONTS Vol. 3	89

Preisänderungen und Irrtümer vorbehalten

amigaland

a. forster
feldbergplatz 11
6500 mainz

Bestellservice:

06241/78569

Alle Preise sind Endpreise.
Mindestbestellwert DM 25,-
Versand ausschließlich per
UPS gegen Nachnahme
oder Vorkasse (Scheck, bar).

ler bereits bei der Übersetzung erkannt werden, können sich in den Assemblerquellcode sehr schnell Fehler einschleichen, die später zum Absturz des Systems führen. Eine Fehlersuche durch gedankliches Nachvollziehen der einzelnen Befehle ist bei größeren Programmen sehr aufwendig. Als weiteres Bonbon liefert Hisoft daher mit dem Debugger »Monam« ein Programm, mit dem übersetzte Objektprogramme im Einzelschrittmodus mit laufender Ausgabe der Registerinhalte untersucht werden können. Mit weiteren Funktionen lassen sich die Inhalte von Speicherplätzen oder Register ausgeben oder ändern, ganze Speicherbereiche kopieren, mit einem Wert füllen oder disassemblieren. Durch das Setzen von Unterbrechungspunkten (Breakpoints) läßt sich das Programm an kritischen Stellen gezielt unterbrechen. Da es sich um einen symbolischen Debugger handelt, ist die Angabe von Adressen nicht nur hexadezimal, sondern auch mit den im Programm verwendeten Labeln möglich. Monam arbeitet mit einem viergeteilten Fenster (Bild 1). Neben einem Bereich für Kommandos und Meldungen können so die Prozessorregister und ein Teil des Speicherbereichs hexadezimal und disassembliert gleichzeitig beobachtet werden. Leider kann das zu untersuchende Programm nur beim Start des Debuggers angegeben werden und so ist das Laden eines weiteren Programms während des Betriebes nicht möglich. Das macht den Einsatz von Editor/Assembler und Debugger im Multiprogrammbetrieb nicht sehr sinnvoll.

Schnelle Übersetzung

Ein Veteran auf dem Amiga ist der Seka-Assembler. Sein Konzept unterscheidet sich grundlegend von dem der Konkurrenten. Das verhältnismäßig kleine Programm beinhaltet Editor, Assembler und Linker. Der Editor enthält die wichtigsten Funktionen für die Erstellung eines Quellprogrammes und ist beim Scrollen von Texten nicht gerade der Schnellste. Dafür dauert der Assemblierungsvorgang nur sehr kurze Zeit. Seka holt sich den Quellcode aus dem Speicher, erzeugt je nach gewählter Arbeitsweise Link- oder Maschinencode und legt diesen auch wieder im Speicher ab. Assemblieren von oder auf die Diskette ist nicht vorgesehen. Die Codepuffer können dann als ausführbares Programm oder Linkmodule auf Diskette gespeichert werden. Mit dem Befehl H (How Big) lassen sich Größe und Adresse der verwendeten Speicherbereiche auf dem Bildschirm ausgeben. Erzeugter Linkcode kann in einen speziellen Linkpuffer kopiert werden. Ist dieser Puffer nicht leer, so wird der Inhalt mit dem bei weiteren Assemblierungen erzeugten Code verkettet. Mit Hilfe des integrierten Debuggers läßt sich das Programm starten und mit ähnlichen Funktionen wie beim Devpak auf Fehler untersuchen. Einfacher geht es wirklich nicht.

Komfort mit Makros

Trotz der einfachen Bedienung fallen auch dunkle Schatten auf den Assembler. So akzeptiert Seka den Befehl »MOVEA« nicht und weist die Adressierung des Bedingungscode-Registers mit »CCR« zurück. Bei der Gestaltung der Assemblerdirektiven (EQU,ORG und andere) hielt man sich nicht wie die beiden Konkurrenten an den Motorola-Standard. Eine Anpassung fremder Programme ist damit notwendig.

Ein weiterer schwerwiegender Nachteil für die professionelle Programmierung ist die fehlende Möglichkeit der Einbindung von Quelltexten während der Assemblierung. Die anderen Assembler kennen hierfür die Anweisung »INCLUDE Filename«. Auf deren Disketten findet man auch eine Reihe Include-Files. In diesen Files werden hauptsächlich den Routinen und Variablen des Betriebssystems — und das sind nicht wenig — symbolische Namen gegeben. Assemblerprogramme, die auf das Betriebssystem zurückgreifen, sind so wesentlich einfacher zu programmieren.

Obwohl nur der Assembler von Metacomco das Wort »Macro« im Namen trägt, sind doch alle drei Kandidaten Makroassembler. Makros sind durch »Name MACRO« und »ENDM« eingeschlossenen Textzeilen im Quellprogramm. Taucht bei einer Übersetzung nach der Definition des Makros dessen Name im Quelltext auf, so ersetzt der Assembler den Namen durch die entsprechenden Textzeilen. Bei den getesteten Assemblern können den Makros bis zu neun Parameter übergeben werden. Das Makro INC im folgenden Beispiel simuliert einen Inkrementbefehl:

```
INC: macro   ; Definition
    ADDQ #1,\1
ENDM

INC DO    ; Aufruf
```

Der Assembler setzt beim Aufruf an die mit <\> und einer Ziffer gekennzeichneten Stellen den entsprechenden Parameter ein.

Eine leistungsfähige Makroverwaltung ist eine unschätzbare Hilfe für den Programmierer. Es lassen sich damit Befehlsfolgen aufbauen, die ähnlich komfortabel sind wie Funktionsaufrufe in höheren Programmiersprachen. Beispiel:

```
MOVEWINDOW: macro
  move.l \1,a0
  move.l \2,d0
  move.l \3,d1
  move.l IntuitionBase,a6
  jsr _MoveWindow(a6)
endm
```

Nach der Definition des Makros wird dann mit der Anweisung »MOVEWINDOW Window,#10,#10« ein Intuition-Fenster um jeweils 10 Punkte nach rechts und nach unten verschoben. Das einmalige Definieren eines Makros entlastet fortan den Programmierer von der Überlegung, in welchem der Register die Parameter der Betriebssystemroutinen stehen müssen. Während sich der Amiga unter Umständen bis zum nächsten Reset verabschiedet, wenn Seka bei einem Makroaufruf innerhalb des aufgerufenen Makros einen weiteren Aufruf findet, können seine Konkurrenten Makros bis zu 8 und 10 Ebenen tief verschachteln.

Mit der bedingten Assemblierung ist es möglich, den erzeugten Code ohne Änderung des Quellprogramms zu variieren. Eine Anwendung dafür ist das Einfügen von Debugginghilfen bei Testversionen:

```
Upro1:
IFNE DEBUG
  lea Text,a0  ;
  jsr StringOut
; Text ausgeben
ENDC
  ...
  rts  ; Ende von Upro1
IFNE DEBUG
Text: dc.b "Bin im Upro
Eins",0
ENDC
```

Der Pseudoopcode (Pseudooperationscode = kein echter 68000-Operationscode) »IFNE« überprüft den Wert des Symbols DEBUG. Ist dieser ungleich Null, werden die Zeilen bis »ENDC« übersetzt. Eine solche Technik kann sinnvoll sein, um bei auftretenden Fehlern den Programmablauf am Bildschirm zu verfolgen. Funktioniert das Programm einwandfrei, so wird durch Setzen von DEBUG auf Null die Debugginghilfe nicht mehr assembliert.

Während Seka praktisch nur die Pseudoops IFEQ und IFNE kennt, arbeiten die beiden anderen Testkandidaten auch hier nach dem Motorola-Standard, der als Parameter für den »IF« fast alle Bedingungen vorsieht, die auch der 68000 in seinen Branchbefehlen verwendet. Die Leistungsfähigkeit eines Assemblers wird oft an den Möglichkeiten der Makroverarbeitung und der bedingten Assemblierung gemessen. Hier lasssen der Metacomco und das Produkt von Hisoft kaum noch Wünsche offen.

Fazit

Mit dem Devpak von Hisoft erhält der Anwender ein äußerst leistungsfähiges Assembler-Entwicklungspaket. Zusammen mit dem Debugger bietet er einen größeren Funktionsvorrat als die beiden anderen. Sieht man vom fehlenden Debugger einmal ab, so weist auch der Metacomco Makro-Assembler einen ähnlich hohen Qualitätsstandard auf. Für professionelle Projekte sind beide wohl am besten geeignet. Allerdings sollte sich die Preisgestaltung mancher Anbieter des Metacomco deutlich an seinen Leistungen orientieren. Der Seka-Assembler ist wegen der einfacheren Bedienung sehr gut für kleinere Programme oder erste Kontakte mit Prozessor und Betriebssystem geeignet. »Learning by doing« läßt sich hier ohne große Wartezeiten auf Assembler oder Linker vollziehen. Je besser man sich aber mit dem Amiga auskennt, desto eher wird ein Umstieg auf leistungsfähigere Assembler unvermeidlich. (pa)

Anbieter: (Adressen siehe Marktübersicht) Seka: Alphatron, ASH-Computerware, Soyka Datentechnik (149 bis 248 Mark)
Makro-Assembler: Alphatron, Atlantis, DTM Werbung & EDV GmbH, Instant Replay, Interplan, ITC, PDC, Philgerma GmbH, Softwareland, SWS Computersysteme, Video Loft Film (179 bis 398 Mark)
Devpak: Markt & Technik, Hans-Pinsel-Str. 2, 8013 Haar bei München (148 Mark)

CLI-Befehle mit der Maus

Das CLI des Amiga ist leider nicht sehr komfortabel. Mit ExDir, dem Programm des Monats, wird die Arbeit zum Vergnügen. Es wird fast nur mit der Maus gearbeitet.

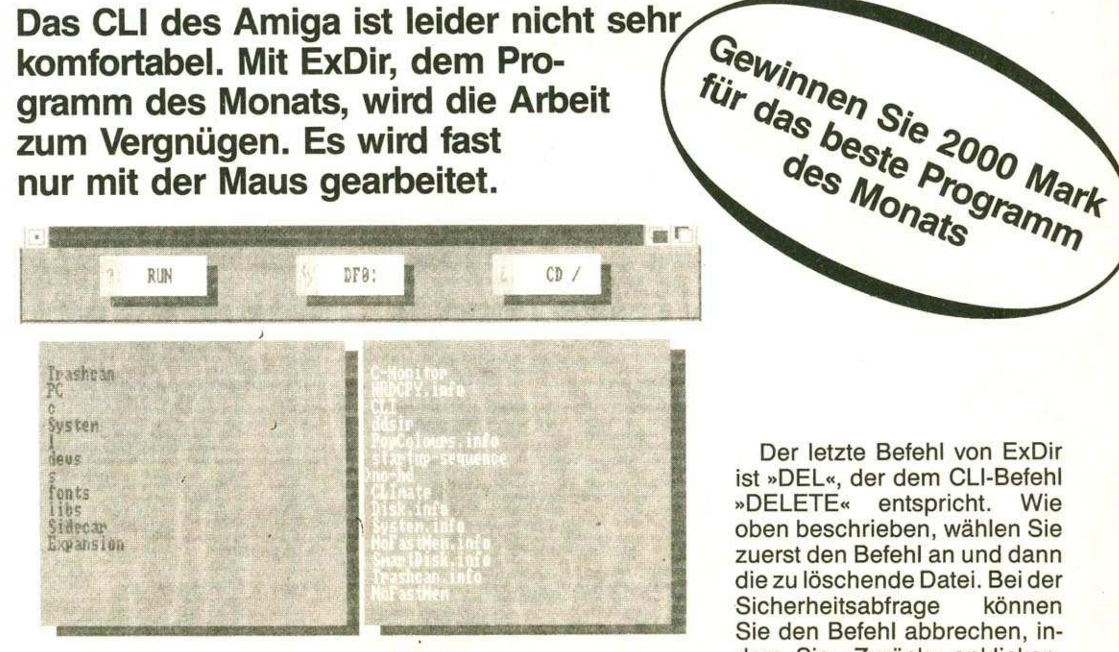


Bild 1. Der Bildschirm von ExDir mit den fünf Blöcken

Die eigentlich positive Eigenschaft des Amiga-DOS, Subdirectories (Dateiunterverzeichnisse) verwalten zu können, wird für den Anwender manchmal zum Ärgernis. Bestimmt wollten Sie auch schon einmal die nutzlosen Dateien von einer Diskette löschen und haben dabei einige Dateien übersehen. In genau solchen Fällen wird ExDir zur unentbehrlichen Hilfe für Sie.

Der Bildschirm von ExDir teilt sich auf in drei kleine und zwei große Blöcke (siehe Bild 1). Im mittleren der drei kleineren Blöcke steht das aktuelle Laufwerk, von dem die Daten gelesen werden. Diese Einstellung können Sie verändern, indem Sie die Maus auf den roten Teil dieses Blocks bewegen und diesen anklicken. Nacheinander erscheinen dann die Laufwerke, die im Programm vorgegeben sind.

In dem linken großen Block stehen die Subdirectories, die im aktuellen Directory vorhanden sind. Den Namen des aktuellen Dateiverzeichnisses können Sie in der linken oberen Ecke des Fensters sehen. Im großen Block rechts daneben stehen die Dateien, welche sich in diesem Subdirectory befinden. Um in ein anderes Dateiverzeichnis zu gelangen, bewegen Sie den Mauszeiger einfach auf den Namen, wobei vor dem angewählten dann eine Markierung in Form eines Größer-Zeichens erscheint. Nach einmaliger Betätigung der linken Maustaste wird dann in den zwei großen Blöcken der Inhalt dieses Subdirectorys angezeigt.

Wenn mehr Dateien vorhanden sind, als auf einmal angezeigt werden können, zeigt sich die Benutzerfreundlichkeit von ExDir. Durch Berühren des oberen oder unteren Randes der großen Blöcke werden die angezeigten Daten nach unten beziehungsweise oben gescrollt und die anderen Dateien oder Dateiverzeichnisse werden sichtbar.

Wollen Sie nun das darüberliegende Verzeichnis sehen, klicken Sie im weißen Teil des rechten kleinen Blocks. Mit dieser Methode gelangen Sie immer in das nächsthöhere Directory, was dem CLI-Befehl »CD /« entspricht. Wollen Sie jedoch das Rootdirectory der Diskette sehen, klicken Sie in den weißen Teil des mittleren kleinen Blocks.

Wenn Sie nun die gewünschte Datei im rechten Block sehen, haben Sie vier Befehle zur Verfügung, die Sie verwenden können. Der aktuelle Befehl steht in dem ganz links stehenden kleinen Block. Die anderen Befehle erreichen Sie durch Anklicken des weißen Feldes. Eine kurze Anleitung zum Programm können Sie lesen, wenn Sie den roten Teil anklicken. Steht nun der erforderliche Befehl in dem Block, gehen Sie mit der Maus einfach auf den Namen der gewünschten Datei und starten den Befehl durch Anklicken des Namens.

Die Befehle

Mit RUN können Sie wie mit dem gleichnamigen CLI-Befehl ein Programm starten, das auf einer beliebigen Diskette steht. Dabei muß es sich nicht unbedingt um Befehle des CLI handeln, es können auch andere Programme gestartet werden. Um ein solches Programm zu starten, klicken Sie so lange auf dem weißen Teil des linken oberen Felds, bis dort »RUN« steht. Nun bewegen Sie die Maus in den großen rechten Block und wählen das Programm aus.

Ein Nachteil des Befehls RUN ist, daß man keine Parameter an das aufzurufende Programm übergeben kann. Bei vielen Programmen muß man aber dem Programm mitteilen, mit welchen Dateien es arbeiten soll. Das fängt schon bei den »normalen« Befehlen des CLI an. Wollen Sie zum Beispiel eine Datei kopieren, müssen Sie angeben, welche Datei Sie wohin kopieren wollen. Die Möglichkeit, Parameter an Programme zu übergeben, bietet ExDir mit dem Befehl RUN &. Dadurch sind Sie in der Lage, auch Compiler und ähnliches von ExDir aus zu starten. Nachdem Sie das Programm ausgewählt haben, erscheint ein Fenster auf dem Bildschirm, in dem Sie hinter dem Befehl die Parameter eingeben können. Löschen Sie den vorgegebenen Text, können Sie hier jedes beliebige Programm ablaufen lassen.

Einen sehr wichtigen Befehl stellt »TYPE« dar, mit ihm können Sie sich den Inhalt von Textdateien anzeigen lassen. Er funktioniert zwar grundsätzlich wie der CLI-Befehl, hat aber einige Vorteile. Mit diesem Befehl können Sie sich den Inhalt von Textdateien ansehen. Zwei Tasten spielen hier eine Rolle: mit <ESC> läßt sich der Vorgang unterbrechen und durch Drücken der Leertaste kann man den Vorgang anhalten. Durch erneutes Betätigen der Leertaste wird dann der Text weiter aufgelistet. Der Vorteil von TYPE ist, daß Sonderzeichen unterdrückt werden, falls Sie eine Programmdatei anzeigen lassen.

Der letzte Befehl von ExDir ist »DEL«, der dem CLI-Befehl »DELETE« entspricht. Wie oben beschrieben, wählen Sie zuerst den Befehl an und dann die zu löschende Datei. Bei der Sicherheitsabfrage können Sie den Befehl abbrechen, indem Sie »Zurück« anklicken, mit »Weiter« wird die Datei dann gelöscht.

ExDir ist erweiterbar

Sollten Sie im Besitz eines Aztec-C-Compilers sein, können Sie ExDir leicht an Ihre Bedürfnisse anpassen. Sowohl die verwendeten Laufwerke als auch die Befehle sind relativ leicht zu erweitern.

Zuerst zum Einbinden von weiteren Laufwerken in ExDir. In Zeile 22 finden Sie die Anzahl der Laufwerke, hier tragen Sie bitte die Zahl der verwendeten Laufwerke minus 1 ein. Nun müssen Sie nur noch hinter Zeile 37 den Namen (zum Beispiel DF3: oder DH0:) eingeben, wobei Sie nicht vergessen dürfen, dahinter ein Komma einzusetzen.

Die Erweiterung von ExDir um Befehle ist verständlicherweise nicht ganz so einfach und erfordert einige Kenntnisse in der Programmiersprache C. Wie bei den Laufwerken muß zuerst die Anzahl der vorhandenen Befehle in Zeile 21 verändert werden. Dann muß eine Prozedur geschrieben werden, die den gewünschten Vorgang durchführt. Sie kann ab der Zeile 349 eingesetzt werden. Damit das Programm auch weiß, wann diese Prozedur aufgerufen werden soll, müssen Sie hinter Zeile 750 noch einen Programmteil einfügen, der Ihr Unterprogramm dann aufruft. Am einfachsten ist es natürlich, Befehle des CLI

zu verwenden, die auf der Diskette im Subdirectory c stehen. Wie eine solche Erweiterung ausschauen kann, sehen Sie in Listing 2.

Damit ExDir überhaupt lauffähig ist, müssen auf der Startdiskette einige Dateien vorhanden sein. Dies sind die CLI-Befehle ASSIGN, MAKEDIR, RUN, DELETE und COPY. Außerdem muß der RAM-Handler im Subdirectory »l« vorhanden sein, da ExDir die RAM-Disk verwendet.

Wenn Sie Erweiterungen zu ExDir geschrieben haben, schicken Sie sie uns zu, wir werden sie dann veröffentlichen. (Arno Gölzer/rb)

Programmname:	ExDir
Computer:	A500, A1000, A2000 mit 512 KB
Sprache:	C
Compiler:	Aztec C, Version 3.20a oder später
Aufrufe:	cc -s +l exdir ln exdir.o -lc32
Bemerkung:	Die Warnung beim Linken kann ignoriert werden

```
1  #include <stdio.h>
2  #include <ctype.h>
3  #include <exec/exec.h>
4  #include <exec/types.h>
5  #include <intuition/intuition.h>
6  #include <libraries/dosextens.h>
7
8  #define ymin   64
9  #define ymax   176
10 #define lmin   25
11 #define rmin   330
12 #define breite 280
13 #define gh     18
14 #define gb     100
15 #define gy     14
16 /*
17    Bei Erweiterungen:
18    befmax = Anzahl der Befehle-1
19    lwmax  = Anzahl der Laufwerke-1
20 */
21 #define befmax  3
22 #define lwmax   2
23
24
25 unsigned char *verzeichnis[100];
26 unsigned char *filename[200];
27 unsigned char kommando[132];
28 unsigned char UNDO[132];
29 unsigned char aktuellDir[100];
30 unsigned char store[100];
31 unsigned char Dblock[512];
32
33 static char *dirname;
34 static char *laufwerk[]={
35     "DF0:",
36     "DF1:",
37     "DF2:",
38 /* Erweiterungen (Laufwerke) hier einfügen */
39 };
40
41 static char *befehl[]={
42     "RUN",
43     "RUN &",
44     "TYPE",
45     "DEL ",
46 /* Erweiterungen (Befehle) hier einfügen */
47     };
48
49 static int zl,zr,y,y1,nr,slinks,srechts,befnr,first;
50 static int lasty = 20;
51 static int lastx =  5;
52 static int lwnr  =  0;
53 static int flag  =  0;
54
55 USHORT pairs1[] = {0,0,  80,0,  80,11, 0,11, 0,0};/* paarweise XY Koord.  */
56 USHORT pairs2[] = {0,0, 324,0, 324,11, 0,11, 0,0};/* fÜr Gadgets Rahmen   */
57 /* Text in oder Über Gadgets */
58 struct IntuiText ok={1,2,JAM1,16,2,NULL,(UBYTE *)"Weiter",NULL};
59 struct IntuiText zu={1,2,JAM1,16,2,NULL,(UBYTE *)"ZurÜck",NULL};
60 struct IntuiText na={1,2,JAM1,80,-12,NULL,(UBYTE *)"Parameter eingeben:
   ",NULL};
61 /* Vereinbarung - Parameter wird mit max. 130 Zeichen in 'kommando'
62    gespeichert. */
63 struct StringInfo infoN = {kommando,
   UNDO,0,131,0,0,0,0,0,0,0,0,NULL};
64 /* Daten wie Ort, Farbe usw. fÜr Gadgets; mit Zeiger auf
   Rahmenkoordinaten */
65 struct Border border1 = {-1,-1,1,2,JAM1,5,pairs1,NULL};
66 struct Border border2 = {-1,-1,1,2,JAM1,5,pairs2,NULL};
67 struct Gadget gadOK = {
68    NULL,          /* Zeiger auf nächstes Gad */
69    50,40,80,10,       /* links, oben, Breite, Höhe */
70    GADGHCOMP,       /* Flags (hier Complement bei Klick) */
71    GADGIMMEDIATE | RELVERIFY,/* Activationflags */
72    BOOLGADGET,        /* Type */
73    (APTR)&border1   ,   /* Borderstruktur */
74    NULL,      /* Grafik im Gadget */
75    &ok,      /* Gadgettext */
76    0,        /* wird z.Z. nicht benutzt */
77    NULL,     /* Data Struktur fÜr Proportional Gadget */
78    2,        /* Gadget ID (wichtig fÜr Auswertung) */
79    NULL      /* User Data (wird von Intuition ignoriert) */
80 };
81 struct Gadget gadZUR = {
82    &gadOK,198,40,80,10,GADGHCOMP,GADGIMMEDIATE |
      RELVERIFY,BOOLGADGET,
83    (APTR)&border1,NULL,&zu,0,NULL,1,NULL
84 };
85 struct Gadget gadN = {
86    &gadZUR,5,20,324,10,GADGHCOMP,
      RELVERIFY+STRINGCENTER,STRGADGET,
87    (APTR)&border2,NULL,&na,0,(APTR)&infoN,0,NULL
88 };
89 struct NewWindow FensterVereinbarung = {
90    0,0,640,200,               /* xmin,ymin,xmax,ymin   */
91    1,2,                 /* weiss und schwarz   */
92    CLOSEWINDOW | MOUSEBUTTONS,     /* Info's über Maus und
93                                    Closegadget */
94    WINDOWCLOSE | WINDOWDEPTH | ACTIVATE,  /* Closegadget, vor/zurück
95                                         Gadget, Window aktiv   */
96    NULL,              /* keine (Intuition) Gadgets */
97    NULL,              /* keine Zeichen für Menüpunkte */
98    NULL,              /* (noch) kein Titel */
99    NULL,              /* aktueller Screen */
100   NULL,              /* keine Bitmap */
101   0,0,0,0,      /* keine Min- oder Maxwerte für das Fenster, da das
102                 WINDOWSIZING-Flag nicht gesetzt wurde */
103   WBENCHSCREEN,           /* der Screen für das neue Window */
104 };
105 struct NewWindow EingabeVereinbarung = {
106   154,75,334,55,-1,-1,
107   MOUSEBUTTONS | GADGETDOWN | GADGETUP,
108   REPORTMOUSE | ACTIVATE | RMBTRAP,
109   &gadN,NULL,NULL,NULL,NULL,
110   0,0,0,0,
111   WBENCHSCREEN,
112 };
113
114 void *ViewPortAddress;
115 struct GfxBase          *GfxBase;
116 struct IntuitionBase *IntuitionBase;
117 struct IntuiMessage  *nachricht;
118 struct Window        *fenster, *eingeben;
119 struct ViewPort      *VP;
120 struct RastPort      *RP1, *RP2;
121 struct FileLock      *dir;
122 struct FileInfoBlock *eintrag;
123 struct FileHandle    *Fhandle, *nwn;
124
125       /*******************************************
126              Funktionen
127        *******************************************/
128
129 aufruf()
130 /* Falls das Programm falsch aufgerufen wurde */
131 {
132 /*
133    \033    =  ESC
134    ESC7   =  Blitz
135    ESC[0m =  normale Schrift
136    ESC[3m =  kursiv
137    ESC[4m =  unterstrichten
138    ESC[7m =  revers
139 */
140
141    printf("\033[0m\033%c\n\033[3m\033[4mAufruf:\033[0m ExDir n\n",7);
142    printf("        n=0 => DF0:\n");
143    printf("        n=1 => DF1:\n");
144    exit(0);
145 }
146
147 keinC()
148 /* Fehlermeldung, falls Fehler mit Ramdisk oder CLI-Befehle fehlen */
149 {
150    printf("\033[0m\033%c\033[3mAchtung mir fehlen CLI-Befehle -
      C(Dir)?\noder klappt die Ramdisk nicht???\033[0m\n",7);
151    exit(0);
152 }
153
154 openw()
155 /* Fenster öffnen, wenn alles klappt - sonst gibt's eine
   Fehlermeldung */
156 {
157    if(!(IntuitionBase = OpenLibrary("intuition.library",0L))){
158       printf("IntuitionLib FEHLER\n");
159       closew();
```

Listing 1. Das C-Listing von »Ex-Dir« (bitte ohne Zeilennummern eingeben)

GO AMIGA!

ABSOFT
- AC Basic.......... 377

ACCESS SOFTWARE
- * Tenth Frame....... 78

ACCESS ASSOCIATES
- Alegra 512 K RAM . 732
- Alegra 2 M RAM . 1446

ACCOLADE
- Mean 18 Golf 59

AEGIS DEVELOPMENT
- Aegis Draw plus . 502
- Aegis Animator .. 271
- Aegis Images 78
- Aegis Sonix 155
- Aegis Diga! 155
- * Arozok's Tomb ... 97

AMISOFT
- Diskzine Nr. AA-AG 29

ANAKIN RESEARCH
- The Easel 963

APPLIED VISIONS
- Future Sound 338

ARRAY's INC.
- Master AM DOS Buch 33

ARTWORX
- Link Word German 58

ASSOCIATED COMPUTER
- Brush Works 58

BANTAM
- AM Manual Buch 49
- Express DOS Refdisk 58
- AM User Guide Buch 37

BAUDVILLE
- Video Vegas 68

BETHESDA SOFTWORKS
- Gridiron 116

C LTD.
- aMEGA 1 M RAM .. 1062
- HARDDISK 20 M .. 1930
- TIMESAVER 155

CAPILANO
- Logicworks 193

CENTRAL COAST SOFTWARE
- DOS 2 DOS 107
- Disk 2 Disk 97

CLASSIC IMAGE
- Diablo 59

COMPUMED
- Mirror Koppierer 97
- Mirror Hacker Pak 97

CREATIVE MICRO
- Kick Eliminator . 251

CRYSTAL ROSE
- * Analytic Art 116

DIAMOND SOFTWARE
- C - Monitor V 2.0 179

DIGITAL CREATIONS
- Gizmos 2.0....... 135
- * D'Buddy 155
- Digital Link 135

DIGITAL SOLUTIONS
- * LPD Writer 251
- * LPD Filer 251
- * LPD Planner 251

DISCOVERY
- Marauder II 78
- Grabbit 58
- * Key Genie 97

EAGLE TREE
- Butcher IFF Util. 72

ECE RESEARCH
- ECE Midi Interface 116
- * Clip 126

ELECTRONIC ARTS
- Deluxe Paint 2 .. 251
- Paint Data 1 2 3. 58
- Deluxe Video 2 .. 251
- Deluxe Music 2 .. 193
- Music Data 1 58
- Adventure Construct 78
- * Autoduel 97
- Marble Madness .. 69
- Chessmaster 87
- Bards Tale 97
- Kings Quest 1 2 3 97
- Ogre 58
- Ultima 3 116

EPYX
- Wintergames 69

FIREBIRD
- * Guild of Thieves 87
- * Knight Orc 87

GOLD DISC INC.
- Pagesetter 290
- * Pagesetter Pro .. 765
- Gold Spell 87
- Lazerscript 87

GOLDEN HAWK
- Midi Gold 153

GRAFOX OF ENGLAND
- Logistix englisch 288

IMPULSE
- Prism 135
- * Proclip 58

INFINITY SOFTWARE
- Grand Slam Tennis 97
- Galileo 193

INNOVATIVE TECHNOLOGY
- Diskmappe 19

INOVATRONICS
- Power Windows ... 174

INTERACTIVE SOFTWORKS
- * The Calligrapher 193

ISM INC.
- The Surgeon 116

JAGWARE INC.
- Alien Fires 78

JUMPDISK
- Diskmagazin 19

LATTICE
- C Compiler V 3.1 395

MANX SOFTWARE
- C Commercial ... 963
- C Developers ... 578

MERIDIAN SOFTWARE
- Zing 155
- Games Gallery .. 58
- Zing Keys 97

METACOMCO
- Shell CLI Deluxe 135
- Assembler 193
- Toolkit 97

MICHIGAN SOFTWARE
- Insider - 1 M RAM 676

MICRO ILLUSIONS
- * Dynamic Word ... 290
- Dynamic CAD 965
- * Music X Pro 572
- Fairy Tale Adv . 97
- * Land of Legends 97

MICROPROSE
- * Gunship 79

MICROSEARCH
- * City Desk 290

MICROSMITHS
- TxEd 78
- Fast Fonts 78

MICROSOFT PRESS
- The Amiga Buch . 39

MICRO SYSTEMS SOFTWARE
- Analyze V 2.0 .. 290
- BBS Bulletin Board 193
- Flipside 116
- Online 135
- Scribble plus .. 193
- Organize 193

MILES COMPUTING
- Quintette 87

MIMETICS (SOUNDSCAPE)
- Pro Studio 288
- Midi Interface . 95
- Sound Sampler .. 192

MINDSCAPE
- Deja Vue 97
- Uninvited 97
- Halley Project . 69
- Balance of Power 97
- Sindbad 85
- SDI 85
- * King of Chikago 97

NEW HORIZONS
- Flow 193
- Pro Write 242

NEW-TEK
- Digi View 2 386
- * Digi Paint 116

NORTHEAST SOFTWARE
- Publisher 386
- * Order 97

OXXI
- Encore 76
- * Maxiplan plus .. 385
- * Wow 76

PACIFIC CYPRESS
- Expander 0K RAM 963
- Expander 2M RAM 1349
- Expander 4M RAM 2505

PAR HOME
- Dimmer 39
- Disk Pro Plus .. 58
- * Express Paint .. 193

- Outline 97

PRECISION SOFTWARE
- Amiga System Buch 31

PROGRESSIVE PERIPHERALS
- CLI Mate 68

PSYGNOSIS
- Brataccas + Arena 99
- * Barbarian 79
- * Terrorpods 79

RAINBIRD
- Jewels of Darkness 69

SLIPPED DISK
- DOS/BASIC Schablone 20

SOFTTOUCH
- Custom Screens .. 135

SOFTWARELAND
- GO AMIGA! Datei . 178

SSI
- Computer Baseball 78
- Kampfgruppe 116

SUBLOGIC
- Flightsimulator 2 97
- * Jet 119
- Scenery Disk 7 .. 49
- AM Inter Cable .. 10

SUNRIZE INC.
- Perfect Sound ... 174
- Studio Magic 78
- * Desktop Artist .. 58

TDI SOFTWARE
- Modula2 Standard 193
- Modula2 Developer 290
- Modula2 Commercial 579
- Examples Disk ... 49
- Amiga Editor 97
- The Grid 97
- Kermit Source ... 58

TAURUS IMPEX
- Acquisition 579

THE OTHER GUYS
- Omega File 155

TRUE BASIC
- Interpreter 193
- Compiler Runtime 290
- Developers Toolkit 97

UNISON WORLD
- Printmaster plus 97
- Art Gallery 1 2.. 58

VERSASOFT
- dBMan 290

VIZA SOFTWARE
- * Vizawrite englisch 290

VIP TECHNOLOGIES
- VIP Professional 290

WAVE PAD
- Mouse Pad 10

WESTCOM INDUSTRIES
- Disk File Organizer 99
- Harddisk Backup . 139

WORD PERFECT
- * Word Perfect 763

ZUMA GROUP
- TV Text 193
- Fonts 1 2 3 68

Bestellservice:

BRD: 0041-1-3115959
CH: 01-3115959

Geschäftszeiten:
10.00–12.30, 13.30–18.30 Uhr, außer montags,
Sa.: 10.00–16.00 Uhr.

Versand ins Ausland nur Vorkasse (Scheck, bar, Visa Card, Master Card) zzgl. DM 7,– Porto.

softwareland
Franklinstraße 27
CH-8050 Zürich (Schweiz)

* = Lieferbar nach Verfügbarkeit.

Lieferungen ins Ausland sind zu deklarieren.

Preisänderungen vorbehalten.

```
160        }
161     if(!(GfxBase = OpenLibrary("graphics.library",0L))){
162        printf("GraphicsLib FEHLER\n");
163        closew();
164        }
165     if(!(fenster = OpenWindow (&FensterVereinbarung))){
166        printf("Kann WINDOW nicht öffnen\n");
167        closew();
168        }
169     VP  = ViewPortAddress;
170     RP1 =  fenster->RPort;
171  }
172
173  closew()
174  /* Fenster schliessen */
175  {
176     if(fenster)
177        CloseWindow(fenster);
178     if(GfxBase)
179        CloseLibrary(GfxBase);
180     if(IntuitionBase)
181        CloseLibrary(IntuitionBase);
182          FreeMem(eintrag,sizeof(struct FileInfoBlock));
183     /* RamDisk löschen c: wieder auf Diskette    */
184     Execute("SYS:c/assign c: SYS:c",0L,0L);
185     Execute("RAM:c/del ram:c all q",0L,0L);
186     exit(0);
187  }
188
189  message(win)
190  /* Nachricht, z.B. Mausklick, empfangen */
191     struct Window *win;
192  {
193      int nklasse=0;
194
195     if (nachricht = GetMsg(win ->UserPort)){
196       nklasse = nachricht ->Class;
197       ReplyMsg(nachricht);
198     }
199     return(nklasse);
200  }
201
202  print(port,text,col, xpos, ypos)
203  /* Ausgabe eines Textes, an einer definierten X/Y Position,
204     in einer bestimmten Farbe                   */
205
206     char *text;
207     int port, col, xpos, ypos;
208  {
209     SetAPen(port, col);         /* Farbe wählen */
210     Move(port, xpos, ypos);          /* X/Y Position */
211     Text(port, text, strlen(text));   /* Text       */
212  }
213
214  strmem(von)
215  /* Platz schaffen für den eben gelesenen Namen */
216
217     char *von;
218  {
219     char *nach;
220
221     if((nach = AllocMem(strlen(von)+1,MEMF_PUBLIC|MEMF_CLEAR)) != NULL)
222         strcpy(nach,von);
223     return(nach);
224  }
225
226  checkxy()
227  /* Hier wird festgehalten, wo sich der Mauszeiger gerade befindet und
228  gegebenenfalls gescrollt oder das Prompt geschrieben */
229  {
230     register x,y,spalte;
231
232          y=(fenster->MouseY);      /* YPosition der Maus */
233     if(y < ymin-17) return();      /* ausserhalb der Blocks*/
234          if (y+4-(y+4)%8 == y1) return();  /* selbe Pos. wie eben */
235     print(RP1,">",0,lastx,lasty);
236               /* Prompt an der eben verlassenen
236               verlassenen Stelle
237               löschen      */
238          x  = (fenster->MouseX);      /* XPosition der Maus */
239          y1 = y+4-(y+4)%8;
240     nr = (y1-ymin)/8;
241          if(y > ymax+8) nr=(ymax-ymin)/8;   /* nr = an
242          wievielter Stelle (von oben)    */
243          if(y < ymin) nr=1;            /* der Name steht   */
244
245          if(x > rmin-8 ){          /* rechts, also geht's */
246          spalte = rmin-8;       /* um Files (Flag=1) */
247          if(y < ymin-1){
248           y1=ymin+8;       /* Wenn der Mauszeiger */
249           srechts--;       /* über dem ersten      */
250           if(srechts+nr < 1){   /* Eintrag steht und      */
251               srechts=0;   /* genug Files gelesen */
252               return();   /* wurden, wird 'runter    */
253               }        /* gescrollt      */
254                 ScrollRaster(RP1,0,-8,rmin,ymin-8,rmin+breite,ymax);
255            print(RP1,filename[nr+srechts],1,rmin,y1);
256            Delay(10);       /* ca. 0,2s warten    */
257            return();
258            }
259           if(y1 > (zr*8+ymin)){       /* Zeiger tiefer, als */
260             flag = 0;       /* Namen gelesen wurden */
261             return();
262            }
263           if(y > ymax+8){       /* Jetzt wird hoch-    */
264            y1=ymax;       /* gescrollt       */
265            srechts++;
266            if(srechts+nr > zr){
267               srechts--;
268               return();
269               }
270                 ScrollRaster(RP1,0,8,rmin,ymin+1,rmin+breite,ymax+8);
271            print(RP1,filename[nr+srechts],1,rmin,y1);
272            Delay(10);
273            return();
274            }
275                 if(checkblock() == 1){/* Zeiger im Block ? */
276             print(RP1,">",3,spalte,y1);
277              lastx=spalte;       /* Prompt schreiben und    */
278             lasty=y1;       /* dessen Position    */
279             flag=1;       /* merken.        */
280            }
281          return();
282          }
283       else {             /* Diesmal sind die    */
284          spalte = lmin-8;       /* Directories an der    */
285          if(y < ymin-1){       /* Reihe, aber ansonsten*/
286           y1=ymin+8;       /* geht's weiter, wie    */
287           slinks--;       /* oben beschrieben    */
288           if(slinks+nr < 1){
289               slinks=0;
290               return();
291               }
292               ScrollRaster(RP1,0,-8,lmin,ymin-8,lmin+breite,ymax);
293          print(RP1,verzeichnis[nr+slinks],2,lmin,y1);
294          Delay(10);
295          return();
296          }
297          if(y1 > (zl*8+ymin)){
298               flag = 0;
299               return();
300              }
301          if(y > ymax+8){
302           y1=ymax;
303           slinks++;
304           if(slinks+nr > zl){
305               slinks--;
306               return();
307               }
308          ScrollRaster(RP1,0,8,lmin,ymin+1,lmin+breite,ymax+8);
309          print(RP1,verzeichnis[nr+slinks],2,lmin,y1);
310          Delay(10);
311          return();
312          }
313          if(checkblock() == 1){
314            print(RP1,">",3,spalte,y1);
315             lastx=spalte;
316            lasty=y1;
317            flag=2;   /* flag=2 bedeutet Directoryname */
318            }
319          return();
320          }
321  }
322
323  cd(newdir)
324  /* ähnlich dem CLI-Befehl CD */
325      unsigned char *newdir;
326  {
327      int  i;
328
329      strcpy(store,aktuellDir); /* 'Sicherheitskopie' */
330      if (*newdir == '/') {   /* CD /                 */
331     for (i=strlen(aktuellDir);aktuellDir[i]!='/' &&
     aktuellDir[i]!=':';i--);
332     if(aktuellDir[i] == ':') i++;  /*  Das  ':' behalten wir
     besser */
333     aktuellDir[i] = '\0'; /* String 'aktuellDir[]'
     abschliessen */
334      }
335      else if (newdir[strlen(newdir)-1] != ':'){ /* nicht  DF0:  oder DF1:*/
336        if (aktuellDir[strlen(aktuellDir)-1] != ':')
337                 strcat(aktuellDir,"/");
338        strcat(aktuellDir, newdir); /* nach evtl.'/' den neuen      */
339      }                    /* Dirname anhängen */
340        else strcpy(aktuellDir, newdir);   /* DF0: oder DF1: */
341     /* nachsehen, ob die neue Directory existiert: */
342     if((dir = Lock(aktuellDir, ACCESS_READ)) != 0) UnLock(dir);
343     else{
```

```
344      /* Neue Directory existiert nicht! */
345       if(meldung(0,aktuellDir)==0) strcpy(aktuellDir,store);
346       }
347       return(aktuellDir);
348   }
349
350  meldung(art,nachr)
351  /* 2. Fenster (mit Gadgets) öffnen. (Fehlermeldung, Parametereingabe
     usw.) */
352     int art;
353     char *nachr;
354  {
355     int gadzeiger;
356     int okgad=0;
357
358     if(art == 0 || art == 2) gadzeiger=&gadZUR;/* nur Weiter und
359                          Zurück - Gadget */
360     else        gadzeiger=&gadN;  /* Gadget zur Parametereingabe */
361
362     EingabeVereinbarung.FirstGadget=gadzeiger;
363     if(!(eingeben = OpenWindow(&EingabeVereinbarung))){
364         printf("\033%c",7);
365         return(0);
366     }
367     RP2=eingeben->RPort;
368     RefreshGadgets(gadzeiger,eingeben,NULL);
369     /* Text im Fenster vereinbaren: */
370     if(art == 0){
371         print(RP2,"Kann File od. Verzeichnis nicht öffnen:",1,15,15);
372         print(RP2,nachr,3,15,25);
373     }
374     if(art == 2){
375         print(RP2,"File wirklich löschen ?",1,15,15);
376         print(RP2,nachr,3,15,25);
377     }
378     /* warten auf KLICK in Gadget Weiter oder ZurÜck */
379     while(okgad == 0){
380       switch (message(eingeben)) {
381           case GADGETUP  :
382           case GADGETDOWN:{
383                   okgad=gadw();
384                   break;
385                 }
386           default:{
387               break;
388               }
389       }
390     }
391     CloseWindow(eingeben);
392     return(okgad-1); /* 1 bei 'weiter', 0 bei 'zurÜck' */
393  }
394
395  gadw()
396  /* feststellen ob 'Weiter-' oder 'ZurÜckgadget' gewählt wurde */
397  {
398     struct Gadget *found;
399     int foundID;
400
401     found  =(struct Gadget *) nachricht->IAddress; /* gewähltes Gad */
402     foundID=found->GadgetID;   /* GadgetID feststellen */
403     if(foundID==1 || foundID==2) return(foundID);
404     return(0);
405  }
406
407  setblocks()
408  /* der weisse Bereich,die Linie und die Schatten
409     der beiden grossen Blocks zeichnen          */
410  {
411
412   SetAPen(RP1, 2);       /*  schwarz      */
413   SetDrMd(RP1, JAM1);      /*  zeichenmodus  */
414   Move(RP1, 0, ymin-20);      /*  cursor setzen  */
415   Draw(RP1, 640, ymin-20);   /*  Linie zeichnen */
416
417   block(1,0,ymin-18,640,200);
418   block(2,lmin+7,ymin-5,lmin+12+breite,ymax+17);
419   block(2,rmin+7,ymin-5,rmin+14+breite,ymax+17);
420  }
421
422  clearblocks()
423  /* die beiden grossen Blocks löschen */
424  {
425   block(0,lmin-8,ymin-10,lmin+breite,ymax+12);
426   block(0,rmin-8,ymin-10,rmin+breite,ymax+12);
427  }
428
429  setbef()
430  /* Block links oben */
431  {
432     block(2,85,gy+5,85+gb,gy+5+gh);      /* Schatten   */
433     block(1,75,gy,75+gb,gy+gh);       /* Block     */
434     block(3,75,gy,95,gy+gh);       /* roter Teil    */
435     print(RP1,"?",0,81,26);
436     print(RP1,befehl[befnr],2,120,26);   /* aktueller Befehl */
437  }
438
439  setlw()
440  /* Block mitte oben */
441  {
442     block(2,270,gy+5,270+gb,gy+5+gh);
443     block(1,260,gy,260+gb,gy+gh);
444     block(3,260,gy,280,gy+gh);
445     print(RP1,"<",0,264,24);
446     print(RP1,">",0,268,28);
447           print(RP1,laufwerk[lwnr],2,305,26);   /* aktuelle Laufwerknummer */
448     }
449
450  setcd()
451  /* Block rechts oben*/
452  {
453     block(2,455,gy+5,455+gb,gy+5+gh);
454     block(1,445,gy,445+gb,gy+gh);
455     block(3,445,gy,465,gy+gh);
456     print(RP1,"",0,451,26);
457     print(RP1,"CD /",2,495,26);
458  }
459
460  checkblock()
461  /* gibt EINS zurück, falls Mauszeiger  YPos innerhalb der grossen Blocks */
462  {
463     int my = (fenster->MouseY);
464     if(my >= ymin+4 && my <= ymax) return(1);
465     else return (0);
466  }
467
468  checkgad()
469  /* ist eines der drei oberen Felder gewählt worden ? */
470  {
471     int ret,nkl;
472
473     if(ret = gad(75)){        /* linkes Feld       */
474       druck(75);       /* Feld 'niederdrücken' */
475       switch(ret){       /* rot oder weiss ?  */
476         case 1:{   /* weiss        */
477           befnr++;/* Befehlsnummer um eins erhöhen  */
478           if (befnr > befmax) befnr = 0;
479           setbef();/* Block links wieder herstellen */
480           return(0);
481           }
482         case 2:{     /*das rote Feld wird gewünscht   */
483           setbef();/* Block links wieder herstellen */
484           helptext();/* Erläuterungen auflisten      */
485           return(0);
486           }
487         }
488     }
489
490     if(ret = gad(260)){       /* mittlerer Block    */
491       druck(260);
492       switch(ret){
493         case 1:{    /* weiss        */
494           first = 0;
495           strcpy(kommando,laufwerk[lwnr]);
496           setlw();
497           return(1);
498           }
499         case 2:{    /* rot           */
500           lwnr++;
501           if(lwnr > lwmax) lwnr = 0;
502           setlw();
503           return(0);
504           }
505       }
506     }
507     if(ret = gad(445)){       /* Block rechts       */
508       druck(445);
509       switch(ret){
510         case 1:{    /* weiss       */
511           setcd();
512           strcpy(kommando,"/");
513           return(1);
514           }
515         case 2:{    /* rot           */
516           setcd();
517           infotext();
518           return(0);
519           }
520         }
521     }
522     return(0);
523  }
524
525  gad(xwert)
526  /* ist der Block mit der angegebenen XPosition gewählt worden ? */
527
```

Listing 1. Das C-Listing von »ExDir« (Fortsetzung)

```
528     int xwert;
529 {
530     int mx=(fenster->MouseX);
531     int my=(fenster->MouseY);
532
533     if(mx >= xwert && mx <= xwert+gb && my >= gy && my <= gy+gh){
534        if(mx <= xwert+20) return(2);
535        else return(1);
536        }
537     else return(0);
538 }
539
540
541 block(col,x1,y1,x2,y2)
542 /* Block zeichnen   */
543
544     int col,x1,x2,y1,y2;
545 {
546     SetAPen(RP1, col);
547     SetDrMd(RP1, JAM1);
548     RectFill(RP1, x1,y1,x2,y2);
549 }
550
551 druck(xpos)
552 /* das gewählte Feld wird 'runtergedrückt'   */
553
554     int xpos;
555 {
556     int nkl;
557     block(0,xpos,gy,xpos+gb+10,gy+gh+5);
558     block(3,xpos+10,gy+5,xpos+gb+10,gy+gh+5);
559     while((nkl=message(fenster)) != MOUSEBUTTONS);
560 }
561
562 type(fname)
563 /* ähnlich dem CLI-Befehl TYPE. Nicht so komfortabel, aber dafür werden die
564    Steuerzeichen für Bildschirmblitz oder Umschalten der Zeichensätze usw
565    unterdrückt.                                      */
566
567     unsigned char  *fname;
568 {
569     char *zeichen;
570       int blocklaenge, zaehler;
571
572     /* wegen der einfachen Ein- und Ausgabe von Zeichen
573        wird ein 'CLI' Fenster wird geöffnet */
574     sprintf(Dblock,"RAW:0/0/640/200/TYPE %s",kommando);
575     if(!(nwn=Open(Dblock,MODE_NEWFILE))) return(0);
576
577     Fhandle = Open(fname, MODE_OLDFILE); /* gewähltes  File  eröffnen  */
578     if (Fhandle == 0){
579        wtext("\n\nKann File nicht öffnen !\n\n< RETURN >");
580        while((zeichen=key())!=0x0d);
581        Close(nwn);
582     }
583          do {
584              blocklaenge = Read(Fhandle, Dblock, 512);/* Block  lesen  */
585              for (zaehler = 0; zaehler < blocklaenge; zaehler++) {
586          if(Dblock[zaehler] == '\n')
587           Write(nwn,&Dblock[zaehler],1);/* RETURN erlaubt */
588          else if(Dblock[zaehler] >= 0x20)      /* < 32 verboten  */
589           Write(nwn,&Dblock[zaehler],1);
590          zeichen=key(); /* wird gerade eine Taste gedrückt */
591          if(zeichen==0x1b){   /* ESC = Ende TYPE */
592           blocklaenge=0;
593           break;
594          }
595          if(zeichen==0x20){   /* Space = Stop TYPE */
596           while((zeichen=key())!=0x20){ /* warten auf Space */
597              if(zeichen==0x1b){    /* Ende TYPE */
598                 blocklaenge=0;
599                 break;
600              }
601           }
602          }
603              }
604              } while (blocklaenge == 512);/* solange ein ganzer Block gelesen

605                     wurde ...                  */
606        wtext("\n\033[3m\033[2mEof ...\n< RETURN >\033[0m\n");
607        while((zeichen=key())!=0x0d); /* warten auf Return */
608        Close(nwn); /* Fenster schliessen */
609        Close(Fhandle);   /* File schliessen */
610        return();
611 }
612
613 key()
614 /* Tastendruck */
615 {
616     char zeichen;
617
618     if(WaitForChar(nwn,1)==0) return(0); /* keine Taste gedrückt */
619     Read(nwn,&zeichen,1);    /* nach Tastendruck, dessen Wert in */
620     return(zeichen);   /* 'zeichen' speichern   */
621 }
622
623 readDir()
624 {
625    flag    = 0;
626    y       = ymin;
627    y1      = ymin;
628    zl      = 0;
629    zr      = 0;
630    slinks  = 0;
631    srechts = 0;
632
633     clearblocks();                /* grosse Blöcke löschen */
634
635     /* Diskname lesen */
636     eintrag = (struct FileInfoBlock *) AllocMem(sizeof(struct FileInfo-
        Block),0);
637     dir = Lock(dirname,ACCESS_READ);
638     /* existiert das Verzeichnis? */
639     if((dir == NULL) | !Examine(dir,eintrag)) return(meldung(0,dirname));
640     /* Doppelpunkt hinter Diskettenname hängen */
641     if (first == 0){
642       first++;
643       if(!index(eintrag->fib_FileName,' ')){ /* Space im Disknamen
644                      führt zu Fehlern  */
645         strcpy(aktuellDir,eintrag->fib_FileName);
646         strcat(aktuellDir,":");
647         }
648          }
649
650     SetWindowTitles(fenster,aktuellDir); /* Dirname in Titelleiste */
651
652     /* Directoryeinträge und Filenamen des aktuellen Directorys lesen */
653     ExNext(dir,eintrag);
654     while (IoErr() != ERROR_NO_MORE_ENTRIES) {
655       if (eintrag->fib_DirEntryType > 0){         /* Directoryname */
656      zl++;
657      y1+=8;
658      if(y1 > ymax){
659       /* Mehr Dirnamen als Platz im linken unteren Block */
660       y1 = ymax;
661       slinks++;
662       /* um acht Pixel (Zeichenhöhe) nach oben scrollen */
663       ScrollRaster(RP1,0,8,lmin,ymin+1,lmin+breite,ymax+8);
664       }
665
666         verzeichnis[zl]=strmem(eintrag->fib_FileName); /* Dirname
          speichern*/
667         print(RP1,verzeichnis[zl],2,lmin,y1);        /* und schreiben  */
668         }
669      else{      /* Filename */
670       zr++;
671       y+=8;
672       if(y > ymax){
673       y = ymax;
674       srechts++;
675           ScrollRaster(RP1,0,8,rmin,ymin+1,rmin+breite,ymax+8);
676       }
677
678       filename[zr]=strmem(eintrag->fib_FileName);
679       print(RP1,filename[zr],1,rmin,y);
680       }
681        ExNext(dir,eintrag);
682       }
683 return(0);
684 }
685
686 checkmsg()
687 {
688    int nkl;
689    while (flag != 10){          /* solange flag nicht 10 ist ... */
690    if(nkl=message(fenster));{         /* Was liegt an ?       */
691         switch(nkl){       /* Fenster schliessen oder KLICK */
692           case CLOSEWINDOW:{  /* Fenster schliessen        */
693                 closew();
694                    }
695           case MOUSEBUTTONS:{ /* Mausklick links       */
696           if (checkgad() == 1){ /* die drei oberen Blocks */
697             flag = 10;
698             break;
699             }
700           if (checkblock() == 0){ /* ausserhalb der beiden */
701             flag = 0;         /* grossen Blocks gilt's */
702             break;       /* nicht !        */
703             }
704           switch(flag){        /* ohne Flag auch nicht. */
705             case 1:{          /* Filename angeklickt   */
706                /* Warten bis die Maustaste wieder
707                   losgelassen wird         */
708                while((nkl=message(fenster)) != MOUSEBUTTONS);
709                /* aktueller Dirname nach kommando     */
710                      strcpy(kommando,aktuellDir);
711                /* Doppelpunkt oder Schrägstrich
712                   dahinter            */
```

```
713           if(kommando[strlen(kommando)-1] != ':')
714              strcat(kommando,"/");
715           /* Jetzt noch der gewählte Namen  */
716           strcat(kommando,filename[nr+srechts]);
717           switch(befnr){
718              case 0:{    /*    RUN       */
719                 WindowToBack(fenster);
720                 /* Ausführung mit
721                    Rücksprung     */
722                 Execute(kommando,0L,0L);
723                 /* Fenster wieder
724                    vor          */
725                 WindowToFront(fenster);
726                 break;
727                 }
728              case 1:{    /*   RUN &     */
729                 if(meldung(1,"")){
730                   WindowToBack(fenster);
731                   Execute(kommando,0L,0L);
732                   WindowToFront(fenster);
733                  }
734                  break;
735                 }
736              case 2:{ /*    TYPE      */
737                 type(kommando);
738                 break;
739                 }
740                 case 3:{/* DEL */
741                   if(meldung(2,kommando)>0) {
742                 strcpy(store,"RAM:c/del ");
743                 strcat(store,kommando);
744                 Execute(store,0L,0L);
745                 /* dir neu lesen        */
746                 while(readDir()!=0);
747              }
748              break;
749                 }
750        /* Hier werden Erweiterungen eingefügt */
751              default:{
752                 break;
753                 }
754              }/* switch befnr */
755           break;
756           }/* case 1 (switch flag) */
757
758         case 2:{   /* Directoryname  */
759           while((nkl=message(fenster)) != MOUSEBUTTONS);
760           strcpy(kommando,verzeichnis[nr+slinks]);
761           flag = 10;
762           break;
763           }
764
765        default:{    /* nix von allem */
766           flag = 0;
767           break;
768           }
769         } /* switch flag */
770        }           /* case MOUSEBUTTONS */
771     default:{
772        break;
773        }
774        }            /* switch nkl */
775     }            /* if     */
776
777     checkxy();          /* Maus im Auge behalten und '>' malen  */
778
779        }
780 }
781
782 helptext()
783 /* Erläuterungen auflisten */
784 {
785 char *zeichen;
786 sprintf(Dblock,"RAW:0/0/640/200/ExDir Beschreibung");
787 if(!(nwn=Open(Dblock,MODE_NEWFILE))) return(0);
788
789 wtext("\f\n\033[0m\033[3m                          \033[4mErläuterungen
    zu ExDir\n\033[0m\nDer Bildschirm teilt sich in drei kleine und
     zwei große Blöcke auf. Der linke\ngroße Block zeigt Directories, der
     rechte die Files einer Diskette an.\n");
790 wtext("Durch Anklicken des Eintrages können Sie Unterverzeichnisse
     auflisten oder\nProgramme starten. Wenn der Mauspfeil den oberen
     bezw. unteren Rand eines\nBlockes berühren und genügend Filenamen
     geladen sind, scrollt die Anzeige\n");
791 wtext("nach unten bezw. oben.\nJeder der drei oberen Blöcke ist
     wiederum unterteilt in zwei Felder:\nLinker Block - linke Seite   -
     zeigt diese Erläuterung auf.\n");
792 wtext("                  rechte Seite   - Funktionsvorwahl
    \n                                 RUN   - starten eines Prg's
    \n                                 RUN & - Starten eines Programmes
     mit Parameter\n");
793 wtext("                                      TYPE  - Auflisten eines Textes
    \n                                 DEL   - Löschen eines Files\n");
794 wtext("Block mitte - linke Seite    - Auswahl eines Laufwerkes
    \n               rechte Seite   - Inhalt des gewählten Laufwerkes
     listen\n");
795 wtext("Rechter Block - linke Seite   - Versionsnummer und Infos zu
     ExDir\n                 rechte Seite   - zurück zum letzten Unter
    verzeichnis\n\n\033[0m\033[2m\033[3m<RETURN >\n\033[0m");
796
797 while((zeichen=key())!=0x0d);
798 Close(nwn);
799 }
800
801 infotext()
802 /* Infotext auflisten */
803 {
804 char *zeichen;
805 sprintf(Dblock,"RAW:0/0/640/200/ExDir Info's");
806 if(!(nwn=Open(Dblock,MODE_NEWFILE))) return(0);
807
808 wtext("\f\n\033[0m\033[3m\033[4mExDir Version Nr. 1.1\n\n\033[0mWer
     sagt, daß im CLI des Amiga die Maus tot ist ?!?\n\nExDir ist ein
     Hilfsprogramm, das es erlaubt das Directory einer\n");
809 wtext("Diskette in aller Ruhe anzusehen. Unterverzeichnisse und Files
    \nkönnen mit der Maus auf- und abgescrollt und auch ausgewählt\nwerden
    . Man kann Files starten, sich Text ansehen, in Unter-\n");
810 wtext("verzeichnissen auf- und absteigen usw.\nErklärung siehe
     Hilfsseite (Fragezeichen anklicken).\n\n\n");
811 wtext("\n\n                                              A.G.
    \n\n\033[0m\033[2m\033[3m<RETURN >\n\033[0m");
812
813 while((zeichen=key())!=0x0d);
814 Close(nwn);
815 }
816
817 wtext(tm)
818 /* Textausgabe */
819    char *tm;
820 {
821    Write(nwn,tm,strlen(tm));
822 }
823
824 main(argc,argv)
825    int argc;
826    char *argv[];
827 {
828 /*    RamDisk einrichten */
829   printf("\f\n\033[0m\033[3m\033[4mRichte RamDisk ein ...\n\n\033[0m");
830   if(!Execute("SYS:c/makedir ram:c",0L,0L)) keinC();
831   if(!Execute("SYS:c/copy SYS:c/run to RAM:c",0L,0L)) keinC();
832 /*    Execute benötigt RUN !
833       Hier können Kommandos stehen, die nicht direkt in ExDir
        eingebaut sind.
834       Diese werden von Diskette ins RAM: kopiert und dann später mit
835       Execute aufgerufen.
836       Einbinden der Befehle z.B.:                          */
837
838   if(!Execute("SYS:c/copy SYS:c/delete to ram:c/del",0L,0L)) keinC();
839   if(!Execute("SYS:c/assign c: RAM:c",0L,0L)) keinC();
840
841   if (argc >2) aufruf();            /* optional ein    */
842   if (argc == 2){               /* Argument (0/1) */
843      sscanf(argv[1],"%d",&lwnr);          /* übergeben       */
844      if((lwnr <0) || (lwnr >lwmax)) aufruf();
845    }
846
847   dirname=laufwerk[lwnr];/* Argument = Laufwerk  */
848
849   openw();        /* Fenster öffnen */
850   setbef();       /* Block links zeichnen  */
851   setlw();        /*       mitte           */
852   setcd();        /*       rechts       */
853   setblocks();      /* 2 mal unten         */
854   printf("\f");          /* CLI Bildschirm löschen*/
855
856   FOREVER{
857    while(readDir()!=0);      /* Directory lesen und anzeigen */
858      checkmsg();             /* nach Mausklick Befehle ausführen */
859    dirname = cd(kommando);  /* das Directory, das angeklickt wurde,
860    wird zur aktuellen Directory */
861          }
862 }
```

Listing 1. Das C-Listing von »ExDir« (Schluß)

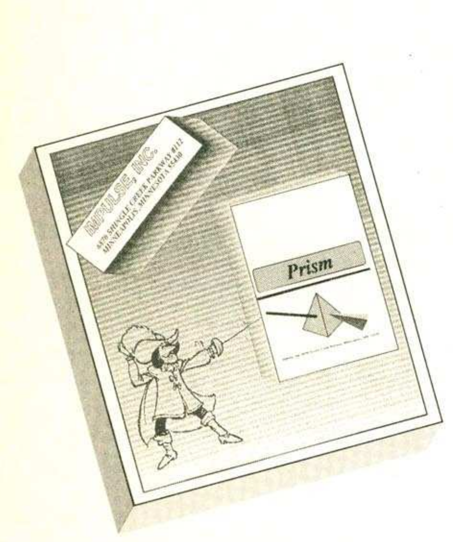
```
case 4:{
       WindowToBack(fenster);
       strcpy(store,"RAM:c/list ");
       strcat(store,kommando);
       Execute(store,0L,0L);
       Delay(50L);
       WindowToFront(fenster);
       break;
       }
```

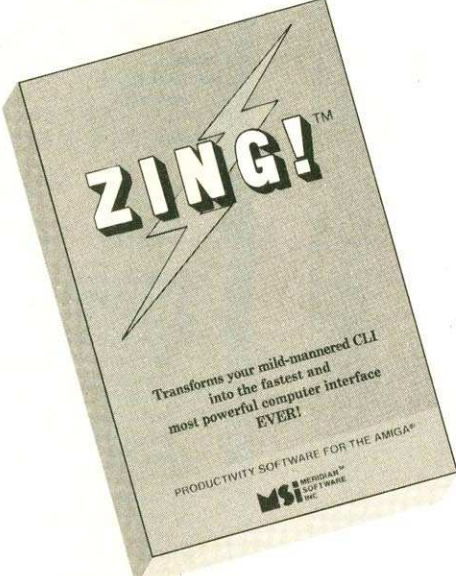
Listing 2. Eine mögliche Erweiterung

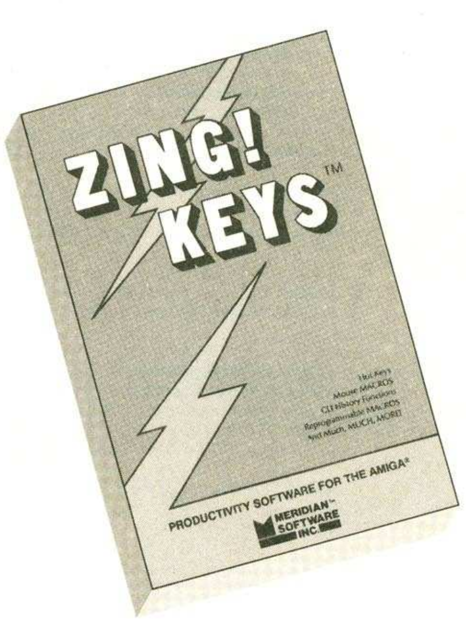
Brandneue Amiga-Software

stop · unentbehrlich für jeden Amiga-User · stop · frisch bei Markt&Technik eingetroffen · stop · deutsche Programmversionen in Arbeit · stop · exklusiv bei Markt&Technik · stop · Update-Service für alle unsere Kunden · stop









Prism

Das einzigartige 4096-Farben-Grafikprogramm

Prism ist nicht nur ein neues Zeichenprogramm unter vielen - Prism ist mehr! Denn Prism beschränkt sich nicht auf die übliche 32-Farben-Palette: Mit Prism haben Sie die ganze Bandbreite der 4096 Farbschattierungen des Amiga zur Verfügung. Auf einmal und in einem Bild! Und Sie können bis zu 4096 neue Farbtöne zu den bestehenden Grafiken hinzufügen und Ausschnitte von einem auf ein anderes Bild übertragen und und und...
Ein einzigartiges Programm für digitalisierte Bilder und deren Manipulation!
Best.-Nr. 52570

DM 159,–* / sFr 142,–/öS 1590,–*

*inkl. MwSt. unverbindliche Preisempfehlung

ZING!

Das mächtige CLI-Werkzeug

Mit ZING! haben Sie endlich das gesamte File-System mit Directories und Subdirectories fest im Griff. Sie beschleunigen mit ZING! alle nötigen Arbeiten mit Files, verwalten bis zu 500 Files und Subfiles und bis zu 100 Directories auf einmal. Die Bedieneroberfläche ist vom Feinsten: Pull-down-Menüs, (Click-)Icons, Funktionstasten. Weitere Optionen wie: Task-Monitor, Printer-Spooler, Screen-Syer/Printer, Screen-Dimmer, Veränderung der Voreinstellung der Funktionstasten und des Systems. Am besten gleich bestellen!
Best.-Nr. 52571

DM 189,–* / sFr 169,–/öS 1890,–*

ZING! KEYS

Ihr ganz persönlicher Amiga

Mit ZING!KEYS machen Sie aus Ihrem Amiga das variable System, das Sie sich schon immer wünschen. Es ist Ihren eigenen Ansprüchen jederzeit anpaßbar! Alle Tasten sind nach Wunsch belegbar: z.B. mit Funktionsaufrufen, Programmaufrufen, Systembefehlen und vorprogrammierten Befehlen. Die Belegung ist natürlich jederzeit abspeicherbar.
Durch die Belegung von »Hot-Keys« haben Sie mit ZING! KEYS ein Multitaskingsystem par excellence!
Best.-Nr. 52572

DM 109,–* / sFr 98,–/öS 1090,–*



Markt&Technik-Produkte erhalten Sie bei Ihrem Buchhändler, in Computerfachgeschäften oder in den Fachabteilungen der Warenhäuser.

707230

Markt&Technik Verlag AG, Buchverlag, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon (089) 4613-0

Bestellungen im Ausland bitte an: SCHWEIZ: Markt&Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Telefon (042) 415656 · ÖSTERREICH: Rudolf Lechner & Sohn, Heizwerkstraße 10, A-1232 Wien, Telefon (0222) 677526 · Ueberreuter Media Verlagsges. mbH (Großhandel), Alser Straße 24, A-1091 Wien, Telefon (0222) 481538-0.

CLI — Die zweite Haut des Amiga Teil 2

Die zweite Anwenderschnittstelle des Amiga, das CLI (Command-Line-Interface), ist ein mächtiges Instrument im Umgang mit diesem Computer. Dieser Kurs deckt die Möglichkeiten auf, die das CLI dem Amiga-Anwender bietet.

Eine Möglichkeit, den Amiga zu bedienen, ist das Arbeiten mit der Workbench. Hier werden alle Operationen mit der Maus erledigt, die Tastatur muß nur noch in seltenen Fällen benutzt werden. Leider lassen sich über die Workbench-Ebene nicht alle Amiga-DOS-Funktionen benutzen. Hierfür ist das auf der Workbench-Diskette zu findende CLI (Command-Line-Interface) das geeignete Werkzeug. Dieser Kurs wird Sie mit allen Finessen des CLI vertraut machen. Nachstehend finden Sie eine Übersicht über die Befehle, die bereits besprochen wurden und in den nächsten Ausgaben behandelt werden.

Kursübersicht:
Teil 1: dir, list, cd, date, endcli
Teil 2: filenote, delete, relabel, rename, type, copy, diskcopy, info, format, install
Teil 3: makedir, ed, prompt, wait, loadwb, setclock
Teil 4: assign, echo, failat, if, say, stack, protect
Teil 5: run, edit, execute, join, lab, quit, skip

Nachdem in der letzten Folge Befehle behandelt wurden, die dem Betrachten des Directories (Inhaltsverzeichnis) der Diskette dienen, setzen wir die Reihe mit Befehlen fort, die gezielt Dateien beeinflussen können. Dieser Kurs läßt sich natürlich dann am besten nachvollziehen, wenn das Gelernte gleich in die Praxis umgesetzt wird. Schalten Sie bitte Ihren Amiga an und legen Sie eine Kopie der Workbench-Diskette ein. Aktivieren Sie anschließend das CLI (im »System«-Directory) durch einen Doppelklick mit der Maus. Sie können nun die Maus aus der Hand legen, da das CLI nur über die Tastatur bedient wird.

Da wir natürlich keine wichtigen Dateien beschädigen wollen, erstellen wir uns eine kleine Testdatei. Geben Sie dazu bitte folgende Befehle ein (die Erklärung folgt später):

```
COPY * TO df0:Test
```

Der Cursor springt nun in eine neue Zeile, liefert aber kein Prompt (>1) mehr. Drücken Sie jetzt zwei- bis dreimal <RETURN> und anschließend <CTRL \> (dies ist der Schrägstrich, der auf der Tastatur in der obersten Reihe rechts vor dem Linkspfeil steht).

Die Floppystation läuft kurz an und erzeugt eine leere Datei mit dem Namen »Test«. Im folgenden werden wir uns bei Diskettenoperationen auf diese Datei beschränken, um keine falschen Dateien zu beschädigen!

Das DOS (Diskettenbetriebssystem) des Amiga erlaubt es, an jede Datei auf der Diskette einen Kommentar von 80 Zeichen anzufügen. Dieser Kommentar kann dazu dienen, bestimmte Informationen zu einer Datei festzuhalten. Beispielsweise, um welche Version des Programms es sich dabei handelt oder welche Dateien noch zu dem Programm gehören. Der Kommentar läßt sich mit dem List-Befehl (siehe Folge 1) betrachten. Dabei wird in einer zusätzlichen Zeile der Kommentar, falls vorhanden, auf dem Bildschirm ausgegeben. Der Text kann auch von der Workbench-Ebene aus betrachtet werden, und zwar mit dem Kommando »Info« aus dem ersten Pull-Down-Menü.

Der Kommentar verfügt über einige Eigenschaften, die möglicherweise nicht sofort verständlich sind. So wird er beispielsweise beim Kopieren einer Datei nicht mitübertragen. Die kopierte Datei, die eigentlich ein identisches Abbild der ersten Datei sein sollte, enthält also keinen Kommentar!

Ein weiterer Punkt ist folgender: Wird eine Datei mit einem neuen Inhalt überschrieben, bleibt der Kommentar der alten Datei erhalten (auch wenn dieser nun nicht mehr zutrifft).

Doch nun zum eigentlichen Befehl, der das Anlegen eines Kommentars ermöglicht: dem Befehl »Filenote«. Mit diesem Kommando kann ein Text zu einer Datei eingegeben werden. Der Text kann aber mit diesem Befehl nicht angesehen werden. Dies ist nur mit dem List-Befehl machbar.

Versuchen wir gleich einmal, eine Datei mit einem Kommentar auszustatten. Vergewissern Sie sich zuerst, daß die eingelegte Diskette nicht schreibgeschützt ist!

Als Beispiel wollen wir der Datei »Test« einen Kommentar anfügen. Der Befehl dazu lautet folgendermaßen:

```
Filenote Test "Kommentar"
```

Dabei sollte der Kommentar immer in Anführungszeichen gesetzt werden, damit auch Leerzeichen oder Sonderzeichen korrekt übernommen werden. Kontrollieren wir gleich, ob der Befehl korrekt ausgeführt wurde. Geben Sie nun ein:

```
List Test
```

Auf dem Bildschirm sollten jetzt die vom List-Befehl her bekannten Ausgaben erscheinen. Zusätzlich findet sich in einer neuen Zeile, durch einen Doppelpunkt eingeleitet, der Kommentar, den wir eben angefügt haben.

Der Text kann auch wieder gelöscht werden. Dazu ist nur anstelle des Kommentars ein leerer String zu übergeben. Nach Eingabe von

```
Filenote Test ""
```

ist die Datei wieder vom Kommentar befreit. Prüfen Sie dies mit

```
List Test
```

nach. Die Kommentarzeile, die vorher durch einen Doppelpunkt eingeleitet wurde, ist nun verschwunden.

Ein weiterer, wichtiger Befehl des CLI ist das Kommando »Info«. Der Befehl gibt Auskunft über die angeschlossenen Disketten- und Festplattenlaufwerke. Nach Eingabe des Befehls

```
Info
```

wird ein Statusreport über alle angeschlossenen Laufwerke und die RAM-Floppy ausgegeben (Bild 1).

Dieser Befehl dient hauptsächlich dazu, einen schnellen Überblick über die momentan verfügbaren Disketten- und Festplattenspeicher sowie deren Kapazitäten zu geben. Er eignet sich auch zum Einbinden in die »Startup-Sequence«. In diese Datei eingebaut, erhält der Anwender gleich beim Start des Amiga die Information über die angeschlossenen Laufwerke angezeigt (wie der Befehl in die »Startup-Sequence« implementiert werden kann, erfahren Sie in einem der folgenden Kursteile).

Die nächsten Befehle, mit denen wir uns befassen, nehmen direkten Einfluß auf Dateien oder die gesamte Diskette. Diese Befehle bewirken in den meisten Fällen eine direkte Veränderung einzelner Bytes, Dateien oder der kompletten Diskette. Sie sind deshalb mit Vorsicht zu genießen, da durch falsche Handhabung leicht eine Datei oder Diskette »zerstört« werden kann.

Der erste Befehl dieser Art, der besprochen wird, dient dem Neu- beziehungsweise Umbenennen einer Datei oder eines Dateiverzeichnisses

```
WB A2000-1.6.87:
2> info

Mounted disks:
Unit Size    Used    Free Full Errs   Status   Name
JH1: 340K       7     671   1%    0  Read/Write JH1
JH0:  20M   14349   26449  35%    0  Read/Write JH0
DF2: 880K    1670      88  94%    0  Read/Write ExtrasD
DF1: 880K    1744      14  99%    0  Read Only  WB A2000-1.6.87
DF0: 880K    1742      16  99%    0  Read/Write Workdisk DM
RAM:   1K       1       0 100%    0  Read/Write

Volumes available:
ExtrasD [Mounted]
WB A2000-1.6.87 [Mounted]
Workdisk DM [Mounted]
RAM Disk [Mounted]
JH0 [Mounted]
JH1 [Mounted]
2>
```

Bild 1. Ein mit »info« erstellter Report

(Subdirectory). Es ist dies der Befehl »Rename«.

Die Schreibweise des Befehls sieht folgendermaßen aus:

```
RENAME alter Name TO
neuer Name
```

Benennen wir doch gleich einmal eine Datei um. Nehmen wir als Beispiel das Programm »Test«, das den neuen Namen »Abfall« erhalten soll. Der Befehl dazu sieht wie folgt aus:

```
RENAME Test TO Abfall
```

Eine Kontrolle mit dem Dir-Befehl ergibt, daß die ehemalige Datei »Test« nun den Namen »Abfall« trägt. Da wir aber den ursprünglichen Namen behalten wollen, benennen wir das Unterverzeichnis wieder um:

```
RENAME Abfall TO Test
```

(Die Angabe von »TO« wäre nicht unbedingt nötig, erleichtert aber die Lesbarkeit des Befehls.) Sollen in einem der beiden Namen Sonderzeichen oder Leerzeichen verwendet werden, so sind die Namen in Anführungszeichen zu setzen. Der Rename-Befehl kann aber noch mehr, als nur eine Datei umbenennen! Geben Sie probehalber diesen Befehl ein:

```
RENAME Test TO Trashcan/
Test1
```

Der Befehl mag Ihnen vielleicht etwas merkwürdig erscheinen, doch in dieser Weise wirkt der Rename-Befehl wie eine Move-Anweisung. Wenn Sie sich mit »Dir« das Directory anzeigen lassen, werden Sie feststellen, daß die Datei »Test« nicht mehr zu sehen ist. Sie ist aber nur aus dem Hauptdirectory verschwunden. »Test« befindet sich jetzt mit dem neuen Namen »Test1« im Subdirectory »Trashcan«, wurde also unter gleichzeitiger Änderung des Namens in ein anderes Directory verlagert.

Auch diesen Vorgang wollen wir wieder rückgängig machen. Die Kommandofolge

```
CD df0:
RENAME Trashcan/Test1 TO
Test
```

bringt die Datei wieder ins Hauptdirectory zurück und gibt ihr den alten Namen, wie eine Kontrolle mit »Dir« beweist.

Ein weiteres Kommando zum Ändern eines Namens ist der Befehl »Relabel«. Allerdings können damit keine Dateien oder Subdirectories umbenannt werden. Dieser Befehl wirkt nur auf den Namen einer Diskette. Möchten Sie also den Namen Ihrer Diskette (der auch unter dem Diskettensymbol in der Workbench erscheint) ändern, ist die nachstehende Form zu verwenden (es sind bis zu 30 Zeichen erlaubt):

```
RELABEL Laufwerk: "neuer
Name"
```

Die Laufwerksangabe ist nicht zwingend erforderlich, empfiehlt sich aber der Übersichtlichkeit halber. Auf die Praxis angewendet würde die Befehlsfolge für Ihre im internen Laufwerk eingelegte Workbench-Kopie so aussehen:

```
RELABEL df0: "Meine
Workbench"
```

Zu beachten ist hierbei, daß die Leerzeichen nicht vergessen werden dürfen. Des weiteren muß ein Ausdruck, der Leerzeichen oder Sonderzeichen enthält, in Anführungszeichen gestellt werden. Wenn Sie das obige Beispiel eingegeben haben, trägt die eingelegte Diskette nun den Namen »Meine Workbench«. Dies können Sie leicht überprüfen, wenn Sie den Befehl »Info« eingeben.

Löschen und Kopieren

Von Zeit zu Zeit kommt es vor, daß man einige Dateien oder Unterverzeichnisse auf seiner Diskette löschen möchte, um beispielsweise Platz für neue Programme zu schaffen. Dazu dient der Befehl »Delete«, der aber sehr mit Vorsicht zu genießen ist, da die betreffenden Dateien endgültig verloren sind (es existieren zwar schon einige Möglichkeiten, die Dateien wieder zu retten, doch ist dafür ein genaues Wissen über den Diskettenaufbau sowie der Besitz eines Diskettenmonitors nötig). Die folgenden Beispiele sollten also nicht ausprobiert werden, wenn Sie nicht sicher sind, daß Sie die betreffenden Dateien noch auf anderen Disketten besitzen. Da wir aber mit einer Kopie (!) der Workbench-Diskette arbeiten, sollte uns das nicht weiter stören, da ja noch ein Original vorhanden ist.

Schreiten wir auch hier gleich wieder zur Praxis und versuchen, eine Datei zu löschen. Hier bietet sich das File »Test« an, welches keine »lebensnotwendigen« Daten enthält. Gelöscht wird die Datei mit folgendem Befehl:

```
DELETE Test
```

oder, falls Sie sich in einem Unterverzeichnis oder einer anderen Diskette befinden, mit

```
DELETE df0: Test
```

Vergessen Sie nicht die Leerzeichen zwischen den einzelnen Parametern!

Es können auch mehrere Dateien (maximal 10) mit einem einzigen Befehl gelöscht werden. Dazu sind nur die einzelnen Namen durch ein Leerzeichen getrennt in die gleiche Befehlszeile zu schreiben. Das nächste Beispiel würde fünf verschiedene Dateien löschen:

```
DELETE df0: Test1 Test2
Test3 Test4 Test5
```

Selbstverständlich können auch Pfadnamen oder Masken verwendet werden. Drei weitere Beispiele:

```
DELETE df0: P#?
DELETE df0: Devs/
Keymaps/USA2
DELETE df0: System/
Er#?t
```

Im ersten Beispiel würden alle Dateien gelöscht, die im Hauptverzeichnis von Laufwerk 0 stehen und mit »P« beginnen. Der zweite Fall löscht das Programm »USA2«, das sich im Subdirectory »Devs/Keymaps« befindet. Der dritte Befehl schließlich löscht alle Dateien, die sich im Unterverzeichnis »System« befinden, mit »Er« beginnen und als letzten Buchstaben ein »t« besitzen.

Der Delete-Befehl gibt bei mehr als einer gelöschten Datei eine Meldung aus. Wer diese Meldung unterbinden möchte, kann am Schluß der Befehlszeile »QUIET« oder abgekürzt »Q« eingeben. Dieser Zusatz unterdrückt die Ausgabe der Meldung. Ein Beispiel:

```
DELETE Test1 Test2 Test3
QUIET
```

oder abgekürzt

```
DELETE Test1 Test2 Test3
Q
```

Auch beim Delete-Befehl kann mit Jokern und Maskierung gearbeitet werden (#?, ??).

»Delete« läßt außer »Quiet« noch den Parameter »All« zu, der, am Schluß der Befehlszeile angehängt, folgende Wirkung hat: Wenn Sie beispielsweise die Kommandozeile

```
DELETE df0: De#? ALL
```

eingeben würden, so wären anschließend alle Dateien und auch Subdirectories verschwunden, deren erste beiden Buchstaben mit »De« begonnen hätten. Man kann die Maskierung auch noch weiter treiben. Der Befehl

```
DELETE #?/#?E#?
```

würde sämtliche Dateien auf der gesamten (!) Diskette löschen, die irgendwo in ihrem Namen ein »e« enthalten (auch alle Subdirectories!). Mit diesem Befehl sollte also sehr vorsichtig umgegangen werden, da man leicht Dateien unrettbar löschen kann, die noch gebraucht werden. Außerdem erfolgt vor der Befehlsausführung keine (!) Sicherheitsabfrage, ob die Dateien wirklich entfernt werden sollen.

Eines ist beim Delete-Befehl noch anzumerken: Es kann kein Subdirectory gelöscht werden, wenn in ihm noch Dateien enthalten sind. Bevor das Unterverzeichnis gelöscht werden kann, müssen also zuerst alle (!) Dateien aus diesem entfernt werden.

Des weiteren können sich Dateien auch Löschversuchen widersetzen. Jede Datei enthält vier Flags (Schalter), die bestimmte Manipulationen mit der Datei zulassen oder verbieten (siehe Amiga 6/7, Seite 60). Den Zustand dieser vier Flags (»r-w-e-d«) können Sie sich auch mit dem List-Befehl ansehen. Nur wenn das »d«-Flag gesetzt ist, läßt sich die Datei entfernen. Ändern lassen sich diese Flags mit dem »Protect«-Befehl, der im vierten Teil des CLI-Kurses besprochen wird.

Vorbereiten einer Diskette

Um Daten auf eine Diskette speichern zu können, ist nötig, daß die betreffende Diskette formatiert ist. Formatieren bedeutet in diesem Fall, daß die Magnetscheibe(n) in konzentrische Kreise und die Kreise in einzelne Sektoren eingeteilt werden. Es soll hier nicht näher auf die Aufteilung einer Diskette eingegangen werden. Wichtig ist hier nur zu wissen, daß Disketten auf jeden Fall formatiert sein müssen, bevor sie genutzt, das heißt Daten auf ihnen gespeichert werden können. Beim Amiga dient dazu der Befehl »Format«. Dieser Befehl ermöglicht es allerdings nur, Disketten zu formatieren. Festplatten oder 5¼-Zoll-Laufwerke können mit diesem Befehl nicht vorbereitet werden (auf diesen speziellen Befehl wird in einer späteren Folge eingegangen). Der Befehl benötigt folgende Schreibweise:

```
FORMAT DRIVE Laufwerk:
NAME Name NOICONS
```

Es ist auf die korrekte Schreibweise zu achten. Leerzeichen dürfen nicht vergessen werden! Außerdem sind die Formalparameter »DRIVE« und »NAME« immer mit anzugeben. Der Parameter »NOICONS« kann, muß aber nicht eingetippt werden. Läßt man ihn weg, installiert das DOS auf der Diskette das Subdirectory »Trashcan« und das zugehörige Icon (Symbol). Gibt man dagegen »NOICONS« an, werden die »Trashcan«-Dateien nicht erzeugt. Um beispielsweise eine Diskette im externen Laufwerk (df1) zu formatieren, müßte folgende Befehlszeile eingegeben werden (versuchen Sie nicht, diesen Befehl nachzuvollziehen, wenn Sie sich nicht sicher sind, ob Sie die Daten in dem betreffenden Laufwerk noch benötigen. Der Amiga verlangt zwar zuerst eine Sicherheitsbestätigung, doch wie leicht kann man sich vertippen... Nach der Formatierung sind diese Daten unrettbar (!) verloren!):

```
FORMAT DRIVE df1: NAME
"Leere Diskette"
```

Auch hier muß der letzte Ausdruck wieder in Anführungszeichen gesetzt werden, wenn Leerzeichen oder Sonderzeichen enthalten sind (zulässig sind insgesamt 30 Zeichen). Wenn die Diskette nicht schreibgeschützt war, besitzt sie nun den Diskettennamen »Leere Diskette« und kann beschrieben werden.

Eine so vorbereitete Diskette kann in diesem Stadium allerdings nur zur reinen Datenspeicherung verwendet werden. Wie Sie sicher wissen, verlangt der Amiga nach dem Einschalten beziehungsweise Laden des Kickstarts eine Workbench-Diskette. Der Amiga würde die eben formatierte Diskette aber nicht als Startdiskette anerkennen, sondern solange nach einer Workbench-Diskette verlangen, bis eine mit einem Startsektor versehene Diskette eingelegt wird. Dies erledigt der nächste Befehl: »Install« (nicht bei Festplatten oder 5¼-Zoll-Laufwerken).

Um beispielsweise eine Diskette in Laufwerk #1 (df1) startfähig zu machen, reicht die Eingabe der nächsten Zeile:

```
INSTALL df1:
```

Das Laufwerk läuft kurz an und installiert den Startsektor. Nach diesem Vorgang akzeptiert der Amiga diese Diskette als bootfähige (Workbench-) Diskette.

Kopieren von Dateien

Jeder Anwender muß von Zeit zu Zeit Dateien von einem Gerät auf ein anderes kopieren. Sei es, um seinen Druckertreiber oder ein anderes Programm auf eine andere Diskette zu übertragen oder, im Fall des Amiga, eine Datei auszudrucken.

Zum Kopieren einer einzelnen Datei dient der Befehl »Copy«. Dieses Kommando dupliziert den Inhalt einer angegebenen Datei auf ein anderes Gerät. Diese können Diskettenlaufwerke, Festplatten, das RAM, Drucker oder im einfachsten Fall der Bildschirm sein.

Besprechen wir also der Reihe nach die verschiedenen Möglichkeiten, wie man mit dem Copy-Befehl Dateien vervielfältigen kann:

Die erste Art bezieht sich natürlich auf diejenigen Geräte, wo dieser Befehl bevorzugt eingesetzt wird: den Disketten- oder Festplatten-Laufwerken und das RAM.

Die Schreibweise dafür lautet allgemein:

```
COPY Quell-Datei TO
Ziel-Datei/-Gerät
```

Erklären wir dies gleich an einem Beispiel: Sie möchten die Datei »Clock« von Ihrer Workbench-Diskette im internen Laufwerk (df0) auf eine zweite Diskette im ersten angeschlossenen Laufwerk (df1) kopieren. Die Datei befindet sich im Hauptdirectory und soll auch auf der zweiten Diskette im Hauptdirectory stehen. Der Befehl dazu würde lauten:

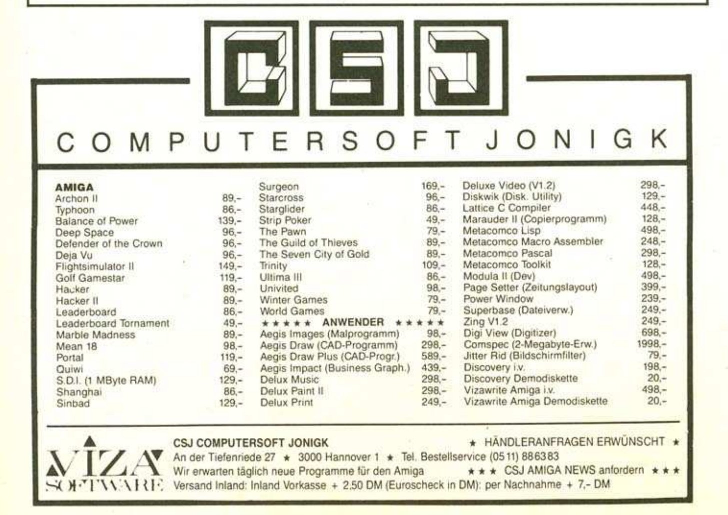
```
COPY Clock TO df1:
```

oder, noch exakter

```
COPY df0:Clock TO df1:
```

(Das »TO« ist nicht zwingend erforderlich, erhöht aber die Lesbarkeit.) Wenn wir davon ausgehen, daß wir uns von Anfang an im Hauptdirectory der im internen Laufwerk (df0) eingelegten befunden haben, sind beide Befehle richtig. Ist aber eine andere Diskettenstation das aktuelle Laufwerk oder befindet man sich in einem Subdirectory, wäre nur





der zweite Befehl richtig (näheres dazu im 1. Teil, Ausgabe 6/7).

Im nächsten Fall soll die gleiche Datei vom selben Laufwerk auf df1 kopiert werden. Die Kopie erhält aber einen anderen Namen. Die Schreibweise:

```
COPY df0:Clock TO df1:
"Neuer Name"
```

Sie können also gleich bei der Befehlseingabe den neuen Namen der Zieldatei festlegen und müssen sie nicht erst später umbenennen. Dies funktioniert auch mit Pfadangeben, um in Unterverzeichnisse zu gelangen. Beispiel: Die Datei »List«, die sich im »C«-Directory von df0 befindet, soll auf df1 in das Unterverzeichnis »XXX« übertragen werden. Gleichzeitig erhält die Datei den neuen Namen »Test«. Die Kommandozeile

```
COPY df0:C/List TO df1:
xxx/Test
```

erledigt diese Aufgabe. Natürlich kann der Copy-Befehl auch maskiert werden (maskieren = Platzhalter einsetzen).

```
COPY df0:C/A#? TO df1:xx
```

Dieser mit Jokern arbeitende Befehl überträgt beispielsweise alle Dateien, die sich im C-Directory von Laufwerk #0 befinden und mit »A« beginnen, in den Ordner »XX« von Laufwerk #1.

Auch beim Copy-Befehl können Sie, wie beim Delete-Befehl, zwei Parameter am Schluß der Befehlszeile eingeben. Der erste ist »QUIET« oder »Q«, der die schon vom Delete-Kommando her bekannte Funktion hat, nämlich die Unterdrückung von Textausgaben auf dem Bildschirm. Beispiel:

```
COPY df0:C TO df1: QUIET
```

Das C-Directory (df0) wird in das Haupdirectory von df1 kopiert. Dabei werden Bestätigungsmeldungen des Copy-Befehls unterbunden.

Der zweite Formalparameter ist das Wort »ALL«. Damit sagen Sie dem Copy-Befehl, daß, falls Sie ein Directory kopieren wollen, auch alle Unterverzeichnisse dieses Ordners mitkopiert werden sollen (normalerweise werden nur die Dateien, nicht (!) die Subdirectories kopiert). Um dies zu verdeutlichen, folgendes Beispiel:

```
COPY df0:Devs TO df1:
```

Es würden alle Dateien übertragen, die sich im Verzeichnis »Devs« befinden, nicht aber die Unterverzeichnisse.

Folgende Befehlszeile würde auch noch die Unterverzeichnisse mitkopieren und gleichzeitig auf der Zieldiskette die entsprechenden Ordner erzeugen:

```
COPY df0:Devs TO df1: ALL
```

Dies würde auch mit Festplattenlaufwerken (dh0/dh1 beziehungsweise beim Amiga 2000 jh0 etc.) und dem RAM funktionieren. Beispiele:

```
COPY df0:c TO dh0: ALL
COPY df1:c TO RAM:
```

Wie weiter oben erwähnt, läßt sich die Ausgabe einer Datei auch auf den Drucker umlenken. Der Drucker würde also eine schriftliche Kopie einer Datei ergeben. Unter Amiga-DOS wird ein angeschlossener Drucker durch »prt:« (für Printer) angesprochen. Legen wir als Beispiel die Ausgabe einer Datei um, beziehungsweise kopieren die Datei auf ein Blatt Papier. Dafür benutzen wir die Startdatei des Amiga (Startup-Sequence), die sich im S-Verzeichnis befindet und von der wir wissen, daß sie lesbaren ASCII-Text enthält. Die Befehlszeile sieht so aus:

```
COPY df0:S/startup-
sequence TO prt:
```

Schon wird ein Ausdruck dieser Datei erzeugt.

Auf diese Weise können beispielsweise auch Daten über die serielle »SER:« oder parallele »PAR:« Schnittstelle gesendet werden.

Ein weiteres Gerät ist der Bildschirm »CON:«.

Wenn Sie, wie auch im Amiga-DOS-Handbuch beschrieben, nachstehende Zeile abtippen, werden alle Eingaben von Tastatur (die Tastatur wird durch den Stern » * « gekennzeichnet) auf ein neu eröffnetes Bildschirmfenster umgeleitet. Sie tippen also etwas ein, das anschließend direkt in einem anderen Fenster, das den Titel »Ausgabefenster« trägt, ausgegeben wird (Abbruch mit <CTRL C> oder <CTRL \>, eingegeben im CLI-Fenster).

```
COPY * TO CON:10/10/200/
100/Ausgabefenster
```

Vielleicht verstehen Sie jetzt auch die Anweisung am Anfang des Artikels, mit der wir die Test-Datei erzeugt haben. Diese Anweisung macht nichts anderes, als Tastatureingaben in eine Datei mit dem Namen »Test« umzuleiten und nach Druck auf <CTRL \> auf Diskette zu speichern.

Ein Befehl, der (wenn keine Kopierprogramme vorhanden sind) das Kopieren kompletter Disketten erlaubt, ist das Kommando »Diskcopy«. Damit lassen sich originalgetreue Kopien von 3½-Zoll-Disketten anfertigen. Kaputte Sektoren oder Tracks werden damit aber nicht übertragen. Diskcopy führt dieselbe Funktion aus, als wenn Sie auf der Workbench-Ebene zwei Diskettensymbole übereinander legen. Dabei ist es nicht nötig, die Zieldiskette vorher zu formatieren. Dies wird von »Diskcopy« erledigt. (Es empfiehlt sich, die Quelldiskette mit Schreibschutz zu versehen.) Das Format des Befehls sieht so aus:

```
DISKCOPY Quell-Laufwerk
TO Ziel-Laufwerk
```

Ein Beispiel: Es soll von der Diskette, die sich im internen Laufwerk (df0) befindet, eine Sicherungskopie auf eine Diskette erstellt werden, die sich in df1 befindet:

```
DISKCOPY df0: TO df1:
```

Nach Eingabe dieses Befehls wird nachgefragt, ob die Diskette wirklich kopiert werden soll. Nach Bestätigung befindet sich nach Abschluß des Vorgangs ein identisches Abbild der Diskette im Ziel-Laufwerk. Die Zieldiskette trägt zur Unterscheidung den Namen »COPY OF xxx«, wobei »xxx« der Name der Ursprungsdiskette ist. Soll die Zieldiskette einen anderen Namen erhalten, kann auch noch der Formalparameter »NAME« am Schluß der Befehlszeile angehängt werden. Dies sieht dann so aus:

```
DISKCOPY df0: TO df1:
NAME "Neuer Name"
```

Die Diskette erhält so gleich beim Kopieren einen neuen Namen.

Es kann auch mit nur einem Laufwerk dupliziert werden. Wenn Sie

```
DISKCOPY df0: TO df0:
```

eingeben, werden Sie (wenn es an der Zeit ist) vom Programm aufgefordert, die jeweils benötigte Diskette (Quelle oder Ziel) einzulegen.

Der letzte Befehl, den wir in diesem Teil besprechen, dient zur Abwechslung nicht der Datei- oder Diskettenmanipulation, sondern »nur« zum Betrachten von Dateien, geeigneterweise ASCII-Dateien, also beispielsweise Textfiles. Es ist dies das Kommando »Type«, das die Inhalte angegebener Dateien auf dem Bildschirm ausgibt. »Type« kann auch wieder in mehreren Variationen verwendet werden. Die einfachste Form ist die Ausgabe einer Datei auf den Bildschirm. Nehmen wir wieder die Datei »Startup-Sequence« im Ordner »S«, von der wir ja schon wissen, daß sie lesbaren Text im ASCII-Format enthält:

```
TYPE S/startup-sequence
```

Der Inhalt des Files wird auf dem Bildschirm angezeigt. Sollte der Text aus dem CLI-Fenster scrollen, hält ein Druck auf die Leertaste die Ausgabe an. <RETURN> oder <BACKSPACE> (der Linkspfeil rechts oben auf der Tastatur) setzt die Anzeige fort. <CTRL C> bricht die ganze Ausgabe ab.

Sie können aber auch die Ausgabe einer Datei auf ein anderes Gerät lenken. Der nächste Befehl gibt beispielsweise die Datei auf einen angeschlossenen Drucker aus:

```
TYPE S/startup-sequence
TO prt:
```

Auch beim Type-Kommando lassen sich Zusatzparameter mitangeben. Der Parameter »OPT N«, am Schluß der Befehlszeile eingegeben, bewirkt, daß vor jede ausgegebene Zeile eine Zeilennummer gesetzt wird (nur für die Ausgabe gültig). Das obige Beispiel, diesmal mit Zeilennummern:

```
TYPE S/startup-Sequence
TO prt: OPT N
```

Der Parameter »OPT H« hat zur Folge, daß die betreffende Datei nicht im Klartext, sondern als Hex-Dump (Ausgabe von Hexadezimal-Zahlen) angezeigt wird. Dies dürfte aber nur für Programmierer von Interesse sein, deshalb soll es hier nur erwähnt werden.

Mit diesem letzten, leicht verdaulichen Befehl schließen wir diesen Teil des Kurses wieder ab. Bedenken Sie, wenn Sie die Befehle ausprobieren, daß Sie nur mit Kopien arbeiten. Schnell kann eine Datei oder Diskette unrettbar zerstört werden!

Übrigens: Auch bei diesen Befehlen hilft ein Fragezeichen, das nach dem Befehl und einem Leerzeichen eingegeben wird, meist weiter. Die CLI-Kommandos zeigen dann kurz die Schreibweise an, die sie erwarten.

Gutes Gelingen bis zum nächstenmal. (dm)

Unterstützende Literatur: Das Amiga-Handbuch, Markus Breuer, Markt&Technik-Verlag, 49 Mark

Basic für Aufsteiger

Teil 1

Der Amiga setzt mit seinen Sound- und Grafikmöglichkeiten neue Maßstäbe. Das Basic des Computers unterstützt die Fähigkeiten der Hardware mit mächtigen Befehlen. Vorbei sind die Zeiten, in denen Kunstwerke mit PEEK und POKE programmiert wurden. Lernen Sie mit diesem Kurs eine neue Qualität der Basic-Programmierung kennen.

Das Basic des Amiga gehört zu einer neuen Generation von Interpretern. Programmzeilen benötigen keine Zeilennummern mehr. Lange Variablennamen machen ein Programm wesentlich lesbarer. Neue Befehle, die Möglichkeit der Zusammenfassung von Programmzeilen zu Blöcken und alphanumerische Ansprungmarken verbessern den Ruf einer Programmiersprache, in der strukturierte Programmierung bisher ein Fremdwort war. Die grafischen Möglichkeiten stellen neue Anforderungen an die optische Gestaltung von Bildschirmausgaben. Es macht einfach mehr Freude, mit einem »schönen« Programm zu arbeiten.

Sie stehen schon mit einem Bein in der Programmierung und können mit Begriffen wie FOR..NEXT oder IF..THEN umgehen? Dieser Kurs hilft Ihnen, festen Fuß zu fassen in einer Umgebung, in der es nicht nur mehr auf Logik, sondern auch auf künstlerische Qualitäten ankommt. Fangen wir deshalb gleich mit einer der großen Fähigkeiten des Amiga an: der Grafik.

Punkt für Punkt

Da alle Theorie bekanntlich grau ist, steigen wir gleich in die Praxis ein. Nach dem Starten von Amiga-Basic sehen Sie auf der rechten Seite des Bildschirms das Listfenster für im Speicher befindliche Programme. Da wir vorerst kein Programm schreiben wollen, schließen Sie bitte dieses Fenster durch Anklicken des Fensterschließsymbols in der linken oberen Ecke.

Den ersten Befehl, den wir nun kennenlernen, ist der elementare Grafik-Befehl überhaupt: das Setzen eines Punktes auf dem Bildschirm. Geben Sie einmal die Anweisung

```
PSET (200,170)
```

ein und beachten nach dem Auslösen mit Return den Effekt auf dem Bildschirm. Dort erscheint in der unteren Hälfte ein Punkt. Die Position des Punktes wird durch die Angaben in Klammern, auch Parameter genannt, festgelegt. Je nach der Einstellung auf Ihrer Workbench-Diskette hat Ihr Bildschirm eine Grafikauflösung von 320 x 200 oder 640 x 200 Punkten. Der Ausdruck 320 x 200 steht dabei für 200 untereinanderliegende Zeilen mit jeweils 320 Punktpositionen (Spalten). Ausgehend von der linken oberen Ecke des Bildschirmfensters haben wir also mit unserem Befehl die Zeile Nummer 170 (das ist die 171. Zeile — die Numerierung beginnt mit 0!) und die 201. Position in der Zeile angesprochen (Bild 1). Man bezeichnet die Angabe der Zeilen- und Spaltenposition auch als X- und Y-Koordinate. Vielleicht ist dem einen oder anderen Leser diese Bezeichnung geläufiger.

Manchmal erfordert der Aufbau einer Grafik auch das Löschen eines einzelnen Punktes. Nach Ausführung der Anweisung

```
PRESET (200,170)
```

verschwindet der soeben gesetzte Punkt vom Bildschirm. Experimentieren Sie ein wenig mit den beiden Befehlen. Sieht man einmal von farbiger Darstellung ab, kann man beliebige Grafiken mit diesen Befehlen erstellen. Ein Beispiel hierfür ist das Ziehen einer Linie durch die folgenden Programmzeilen:

```
Y=100
FOR X=35 TO 180
  PSET (X,Y)
NEXT X
```

Unsere Programmierung nutzt bisher nicht die großen Farbmöglichkeiten des Amiga. Das wollen wir nun nachholen. Einen weiteren Parameter des PSET-Befehls haben wir bisher nämlich verschwiegen: den Farbparameter. Ohne weitere Programmierung lassen sich die vier Farben Blau, Weiß, Schwarz und Rot mit den jeweiligen Farbnummern 0, 1, 2 und 3 darstellen. Diese Farbnummer kann an die PSET-Anweisung angehängt werden. Voreingestellt — das heißt ohne Angabe wirksam — ist hier der Farbwert 1. Die Anweisung

```
PSET (100,150),2
```

setzt also schwarzen Punkt auf blauem Hintergrund (vorausgesetzt, Sie haben die Farbeinstellung ihrer Workbenchdiskette nicht geändert). Unser erstes Beispielprogramm (Listing 1) erzeugt mit PSET und vier Farben interessante Farbmuster.

Sie haben sicher von den fantastischen Farbmöglichkeiten des Amiga gehört. Und jetzt sollen nur vier Farben darstellbar sein? Sicherlich nicht — aber um mit mehr als vier Farben zu arbeiten, sind noch einige Vorbereitungen notwendig.

Je mehr Farben auf dem Bildschirm eingesetzt werden sollen, desto mehr Speicherplatz benötigt der Amiga für dessen Verwaltung. Für einige Anwendungen — beispielsweise die gemischte Ausgabe von Texten und Rechenergebnissen — ist eine große Farbauswahl nicht notwendig. Man hat sich deshalb mit einer Standardeinstellung von vier Farben begnügt.

Für unsere Zwecke ist das zu wenig. Wir definieren uns deshalb einen neuen Bildschirm. Keine Angst — Sie brauchen sich keinen neuen Monitor zu kaufen. Mit dem Amiga kann man sich auf demselben Monitor je nach Erfordernis einen Bildschirm (engl. Screen) mit unterschiedlicher Auflösung und Farbauswahl definieren. Dazu ist die folgende Anweisung notwendig:

```
SCREEN Nummer,Breite,
Höhe,Bitplanes,Auflösung
```

Die »Nummer« dient zur Identifizierung des Screens, »Breite« und »Höhe« legen die Größe des Bildschirms fest. »Bitplanes« ist eine ganze Zahl zwischen 1 und 5 und bestimmt die Anzahl der möglichen Farben (2,4,8,16 oder 32). Mit der »Auflösung«, eine ganze Zahl zwischen 1 und 4, wird die Anzahl der Bildpunkte des Bildschirms mit 320 x 200 (niedrige Auflösung), 640 x 200 Punkten (hohe Auflösung) oder 320 x 400 bzw. 640 x 400 Punkten (Interlace) festgelegt. Wie bereits erwähnt, benötigt ein solcher Bildschirm einen nicht geringen Teil des Speichers im Amiga. Daher ist mit der Anweisung

```
SCREEN CLOSE Nummer
```

die Freigabe des vom Grafikbildschirm belegten Speicherplatzes nach Erfüllung seiner Aufgabe möglich. Damit haben wir nun die Eigenschaften unseres Bildschirms festgelegt.

Einstieg durchs Fenster

Für unsere Ausgabe brauchen wir nur noch ein Fenster (engl. Window) innerhalb dieses Bildschirms. Die Anweisung

```
WINDOW Kennummer,Titel,
(Position der linken obe-
ren Ecke - Position der
rechten unteren Ecke),Typ,
Screennummer
```

sieht komplizierter aus als sie ist. Die einzelnen Parameter kann ein Beispiel am besten verdeutlichen:

```
WINDOW 2,"Test-Fenster",
(50,50-300,150),31,1
```

Wie auch beim SCREEN-Befehl kann man sich mit der Angabe der Kennummer auf das jeweilige Fenster beziehen. Hier sollte jedoch nicht die Nummer »1« verwendet werden, denn Amiga-Basic hat diese für sein Ausgabefenster reserviert. Der Titel erscheint in der Titelleiste des Fensters. Werden keine Koordinaten angegeben, so besitzt das Fenster die Größe des Bildschirms, auf den man sich mit »Screennummer« bezieht. Der »Typ« gibt an, welchen Einfluß Mausbefehle auf das Fenster haben. Sie können damit festlegen, ob das Fenster mit der Maus geschlossen, in den Hindergrund gebracht oder seine Größe verändert werden kann. Die unterschiedlichen Einstellungen schlagen Sie bitte in Ihrem Amiga-Basic-Handbuch nach.

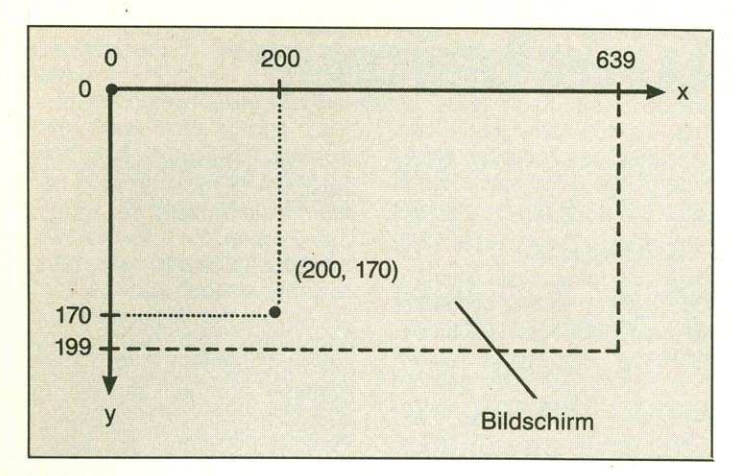


Bild 1. Positionieren eines Punktes mit X- und Y-Koordinaten

Je nach dem für »Bitplanes« gewählten Wert steht Ihnen jetzt eine mehr oder minder große Menge an möglichen Farben zur Verfügung — bis jetzt ist aber noch nirgendwo spezifiziert, welche Farben das sein sollen. Diese Angaben holen wir sofort mit dem Palette-Befehl nach. Die Farben erhalten Kennummern von 0 bis maximal 31 (Einschränkung: In der hohen Auflösung von 640 x 400 sind nur bis zu 16 Farben möglich !). Die Farben auf dem Bildschirm werden im Prinzip genauso festgelegt, wie bei einem gewöhnlichen Fernseher: Aus den drei Grundfarben Rot, Grün und Blau (daher der Name RGB-Monitor) können durch Mischung viele Farben hergestellt werden. Könnte man nur wählen, ob etwa die rote, grüne oder blaue Farbe vorhanden sein soll oder nicht, so gäbe es nur acht verschiedene Farben. Aber es gibt noch die Möglichkeit, die Stärke der einzelnen Farben in 16er-Schritten zu variieren. Dies ergibt für den Amiga eine Anzahl von 16 x 16 x 16 = 4096 darstellbaren Farben. Die Intensitätswerte für eine kleine Auswahl von Farben sind im Basic-Handbuch bei der Beschreibung des Palette-Befehls dargestellt. Die Syntax des Befehls sollte jetzt also keine Fragen mehr aufwerfen:

```
PALETTE Farbnr,Intensität Rot,Int.Grün,Int.Blau
```

Mit dem Befehl

```
COLOR Vordergrund,Hintergrund
```

können Sie bestimmen, welche der mit PALETTE definierten Farben für Vorder- und Hintergrund des Bildschirmes verwendet werden soll. Die Vordergrundfarbe ist dabei diejenige, die beim normalen Schreiben von Texten benutzt wird.

Der einfachste der höheren Grafikbefehle kann mehr als sein Name glauben macht, denn der LINE-Befehl zeichnet außer Linien auch Rechtecke, und die sogar mit einer Farbe gefüllt. Mit der Anweisung

```
LINE (Anfangskoordinaten) - (Endkoordinaten), Farbe
```

zeichnet der Computer eine Gerade von den Anfangskoordinaten zu den Endkoordinaten in der angegebenen Farbe. Wird keine Farbe angegeben, dann wird die aktuelle Vordergrundfarbe benutzt; werden keine Anfangskoordinaten angegeben, wird vom letzten gesetzten Punkt aus gezeichnet.

Aus Ellipsen werden Kreise

Ergänzend kann man noch, durch ein Komma von der Farbe getrennt, entweder den Parameter »B« (Box) oder den Parameter »BF« (Box filled) angeben. In diesem Fall werden die Anfangskoordinaten als linke obere und die Endkoordinaten als rechte untere Ecke eines Rechteckes interpretiert. Bei Angabe von »B« wird nur ein Rechteckrahmen gezeichnet, mit »BF« wird das Rechteck auch noch mit der angegebenen Farbe ausgefüllt. Das Zeichnen von Kreisen wird schon etwas komplizierter. Man kann das schon an der Anweisungssyntax erkennen:

```
CIRCLE Mittelpunktkoordinaten, Radius, Farbe, Start, Ende, Bildverhältnis
```

Bevor wir auf die Feinheiten eingehen, sollte zunächst einmal die Bedeutung der Parameter klar sein. »Start« und »Ende« bezeichnen die Winkel im Bogenmaß (360 Grad = 2 * pi, Zählrichtung entgegen dem Uhrzeigersinn), bei denen das Zeichnen des Kreises begonnen und beendet werden soll. So ist es möglich auch beliebige Kreissegmente zu zeichnen. Das ist ja noch verhältnismäßig einfach. Kommen wir nun zu den Schwierigkeiten: Die Punkte des Amiga-Bildschirms sind keine Quadrate, sondern Rechtecke, deren Seiten im Verhältnis 2.25 : 1 stehen (d.h. das Bildverhältnis beträgt 0.44). Dieser Wert ist auch für das »Bildverhältnis« voreingestellt, doch er gilt nur in der hohen Auflösung mit 640 x 200 Bildpunkten. Arbeitet man dagegen in einer anderen Auflösung, so erhält man keine Kreise, sondern Ellipsen. Dies ist einerseits ganz nützlich, da man so durch die Angabe eines anderen Bildverhältnisses gezielt Ellipsen zeichnen kann, andererseits enthalten Grafiken so in der einen Auflösung Kreise und in einer anderen Ellipsen. Das Programm RGB (Listing 2) versucht diesen Effekt abzufangen, das heißt unabhängig von den gewählten Werten für die Auflösung des Bildes entstehen (meist) Kreise.

Im zweiten Teil von RGB (Listing 2) wird eine weitere Technik der grafischen Fähigkeiten des Amiga demonstriert: das Scrollen (Rollen). Hierunter versteht man die Verlagerung von Bildschirmteilen und dazu stehen uns gleich zwei Möglichkeiten zur Verfügung. Die Anweisung

```
SCROLL (linke obere Ecke)-(rechte untere Ecke), deltax, deltay
```

verschiebt ein durch die Ecken definierten rechteckigen Bildausschnitt um »deltax« Punkte nach rechts und um »deltay« Punkte nach unten. Alle in Schieberichtung aus dem Bildausschnitt hinausgeschobenen Zeilen und/oder Spalten verschwinden damit vom Bildschirm. Durch geschickte Programmierung können mit diesem Verfahren Grafiken über den Bildschirm geschoben werden:

```
FOR I=0 TO 100
SCROLL (50+I,50)-(150+I,150),1,0
NEXT I
```

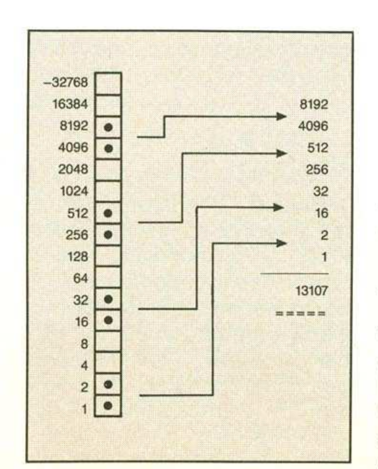


Bild 2. Definition eines Musters für PATTERN-Befehl

Ein Nachteil dieser Technik ist das Löschen derjenigen Teile des Hintergrundes, über die der Bildausschnitt verschoben wird. Ist diese Wirkung nicht erwünscht, empfiehlt sich eine Befehlskombination aus PUT und GET. Überhaupt sind diese beiden Befehle die wohl mächtigsten Grafikbefehle des Amiga-Basic. Um mit diesen Befehlen arbeiten zu können, wird eine Feldvariable benötigt, die genügend Elemente besitzt, um die komplette Bildinformation eines Ausschnitts zu speichern. Daran erkennt man schon, daß dieses Verfahren für ganze Bildschirmseiten wohl nicht geeignet ist. Die Größe des benötigten Integerfeldes läßt sich nach folgender Formel berechnen:
Größe=(Höhe)*INT((Breite+16)/16)* (Anzahl der Bitplanes)+3

Nachdem Sie ein entsprechend großes Feld dimensioniert haben, können Sie mit

```
GET (linke obere Ecke)-(rechte untere Ecke), Feldname%
```

die Bildinformation in die Feldvariable einlesen und mit

```
PUT (linke obere Ecke), Feldname%
```

den Ausschnitt an eine beliebige Stelle des Bildschirms plazieren. Schreiben Sie ihn jedoch wieder an die alte Stelle zurück, so verschwindet überraschend das Bild. Das liegt daran, daß hinter dem Feldnamen beim PUT-Befehl noch ein weiterer Parameter angegeben werden kann, der bestimmt, in welcher Weise das gespeicherte Bild mit dem Bildschirm verknüpft wird. Voreingestellt ist hier der Verknüpfungstyp XOR (Exklusives Oder), der bewirkt, daß alle Bildpunkte, die zugleich im abzubildenden Ausschnitt und im Fenster gesetzt sind, gelöscht werden. Die anderen möglichen Typen für diese Einstellung sind zum einen AND und OR, zum anderen können Sie auch PSET und PRESET angeben. Die beiden letzteren Angaben setzen das Bild ohne Rücksicht auf den Bildschirm-

inhalt, die eine normal, die andere invertiert. Da Sie es nun geschafft haben, Ihre Zeichnung in einer Basic-Variable abzulegen, ist es auch nicht mehr weiter schwierig, diese auf Diskette zu speichern. Setzen Sie bitte in den folgenden Zeilen für den Parameter »Nummer« einen beliebigen Wert von 1 bis 255 und für Dateiname eine beliebige Zeichenkette in Anführungszeichen ein.

```
OPEN Filename FOR OUTPUT
AS Nummer
FOR I = 0 TO (Dimension
des Feldes)
PRINT#Nummer,MKI$
(Feldname%(i));
NEXT I
CLOSE Nummer
```

Nach Ausführung dieser Anweisungen befindet sich Ihre Grafik auf der Diskette. Noch ein Wort zu der Funktion MKI$: Sie wandelt Ihr Argument in einen 2-Byte-String um (MKI$ arbeitet damit entsprechend dem 1-Byte-Befehl CHR$). Der Vorteil dieses Befehls ist, daß die Anzahl der zu speichernden Bytes geringer wird (z.B. 30718 benötigt 5, MKI$ (30718) dagegen nur 2 Byte). Daraus ergibt sich eine schnellere Ladezeit von der Diskette.

Sicherlich wollen Sie die Grafik ja irgendwann auch wieder einlesen.

```
OPEN Dateiname FOR INPUT
AS Nummer
FOR I = 0 TO LOF(Nummer)
\2 - 1
Feldname%(i) = CVI
(INPUT$(2,Nummer))
NEXT I
CLOSE Nummer
```

Die Funktion LOF() ermittelt die Länge der angegebenen Datei in Bytes. CVI() (Convert to Integer) ist die Umkehrung des MKI$-Befehles und die Anweisung INPUT$(2,Nummer) liest aus der eröffneten Datei zwei Byte ein.

Bisher wurden Linien und Flächen immer durchgehend gezeichnet. Das soll sich nun mit Hilfe des Pattern-Befehls ändern. Die Syntax des Befehls ist einfach:

```
PATTERN linie%,flaeche%
```

Dabei ist »linie%« eine normale Integervariable und »flaeche%« ist ein Integerfeld, dessen Dimension eine Potenz der Zahl 2 sein muß (d.h. 1,2,4, 8,16,...). Da die in Basic voreingestellte Dimension 11 ist, müssen Sie diese Variable auf jeden Fall dimensionieren, bevor Sie sie benutzen. Aber denken Sie daran, bei Ihrer Dimensionierung das Element »flaeche%(0)« mitzuzählen.

Wozu sind diese Variablen nun gut? Wenn Sie eine normale Linie ziehen, dann wird dabei jeder Punkt auf dieser Linie gesetzt. Nun können Sie mit Hilfe der Variable »linie%« ganz einfach selbst bestimmen, welche Punkte gesetzt werden und welche nicht. »linie%« enthält eine Zahl, die der Amiga als ein Muster mit 16 Punkten interpretiert. Das Beispiel in Bild 2 zeigt, wie ein solches Muster definiert wird. Jeder der 16 Punkte hat eine bestimmte Wertigkeit. Addieren Sie die Wertigkeiten derjenigen Punkte, die beim Ziehen einer Linie gesetzt werden sollen. Beachten Sie dabei, daß die Wertigkeit des ersten Punktes ein negatives Vorzeichen hat.

Ganz genauso verfahren Sie bei den Einträgen in flaeche%, nur da es sich jetzt um das Füllmuster einer Fläche handelt, ist »flaeche%« eben ein Feld. Wir sind am Ende der ersten Folge unseres Basic-Kurses. In der nächsten Ausgabe werden wir uns mit den Animations- und Sound-Befehlen des Amiga-Basic beschäftigen.

(Jürgen Singer/pa)

```
RANDOMIZE TIMER            : REM   DIESES PROGRAMM ERZEUGT EIN
FOR i = 1 TO 300            : REM   ZUFAELLIGES FARBMUSTER AUF
  PSET(i,1), INT(4*RND)       : REM   AUF DEM BILDSCHIRM
NEXT i
regel(9) = 0 : regel(8) = 0 : regel(7) = 0 : regel(6) = 3
regel(5) = 2 : regel(4) = 1 : regel(3) = 1 : regel(2) = 3
regel(1) = 1 : regel(0) = 0 : REM  REGEL -> 10 ZIFFERN VON 0 BIS 3
FOR j = 1 TO 179
  PSET(  1,j+1),regel(POINT(  1,j) + POINT(  2,j) + POINT(300,j))
  PSET(300,j+1),regel(POINT(299,j) + POINT(300,j) + POINT(  1,j))
  FOR i = 2 TO 299
    PSET(i,j+1),regel(POINT(i-1,j) + POINT(i,j) + Point(i+1,j))
  NEXT i
NEXT j
END
```

Listing 1. Zufallsmuster mit dem PSET-Befehl

```
CLEAR ,20000&              : REM  RGB DEMONSTRIERT DAS MISCHEN
ON BREAK GOSUB beenden     : REM  DER FARBEN ROT, GRUEN UND BLAU
BREAK ON
breite = 320             : REM  50<breite<640 UND 50<hoehe<400
hoehe  = 200               : REM  BEI EINIGEN KOMBINATIONEN WERDEN
aufloesx%=(breite\321)+1   : REM  KREISE NICHT RICHTIG GESCHLOSSEN
aufloesy%=(hoehe\201)*2      : REM  "\" -> GANZZAHLDIVIDISION
modus%    = aufloesx% + aufloesy%      : REM  PARAMETER FUER
scrnx     = aufloesx% * 320              : REM  SCREEN-BEFEHL
scrny     = aufloesy% * 100 + 200          : REM  BESTIMMEN
vh = 4
IF modus% = 1 THEN vh = 2
IF modus% = 2 THEN vh = 1
IF modus% = 4 THEN vh = 2
vh = vh / 2.25                : REM VERHAELTNIS BREITE/HOEHE
ein = 1! : aus = 0
phi = 4*ATN(1!)/3           : REM  WINKEL 60 GRAD
radius = breite
IF breite/vh > hoehe THEN radius = hoehe/vh
radius = radius / 4!
h = SQR(3!) / 2! * radius
IF breite/vh > hoehe THEN h = h * vh
x1 = breite * 7!/24! : y1 = hoehe * 7!/12!
x2 = x1 + radius / vh
IF breite/vh > hoehe THEN x2 = x1 + radius
y2=y1 : x3=(x1+x2)/2 : y3=y2-h
SCREEN 2,scrnx,scrny,3,modus%
WINDOW 2,"RGB",,0,2 : WINDOW OUTPUT 2
PALETTE 0,aus,aus,aus : PALETTE 1,aus,aus,ein
PALETTE 2,aus,ein,aus : PALETTE 4,ein,aus,aus
PALETTE 3,aus,ein,ein : PALETTE 5,ein,aus,ein
PALETTE 6,ein,ein,aus : PALETTE 7,ein,ein,ein
f  = 1 : x  = x1 - 10 : y = y1 + 10
s1 = 2 : s2 = 5 :s3 = 3 : s4 = 4
s5 = 3 : s6 = 4 : GOSUB segment
f  = 2 : x  = x2 + 10 : y = y1 + 10
s1 = 5 : s2 = 0 : s3 = 4 : s4 = 1
s5 = 5 : s6 = 0 : GOSUB segment
f  = 4 : x  = x3       : y = y3 - 10
s1 = 1 : s2 = 2 : s3 = 1 : s4 = 2
s5 = 0 : s6 = 3 : GOSUB segment
f  = 5 : x  = x1       : y = y3
s1 = 1 : s2 = 2 : s3 = 2 : s4 = 3
s5 = 3 : s6 = 4 : GOSUB segment
f  = 3 : x  = x3       : y = y1 + h/2
s1 = 5 : s2 = 0 : s3 = 3 : s4 = 4
s5 = 4 : s6 = 5 : GOSUB segment
f  = 6 : x  = x2       : y = y3
s1 = 0 : s2 = 1 : s3 = 1 : s4 = 2
s5 = 5 : s6 = 0 : GOSUB segment
f  = 7 : x  = x3       : y = y1
s1 = 0 : s2 = 1 : s3 = 2 : s4 = 3
s5 = 4 : s6 = 5 : GOSUB segment
for i=1 to 15000:next i    : REM  WARTEN
REM ************************************ RGB TEIL II
rand = 1
IF breite>hoehe THEN SWAP rand,vh
xl% = x1 - radius / vh   - 5
yl% = y3 - radius * rand - 5 :   REM VERSCHIEBEN
xr% = x2 + radius / vh   + 5 :   REM EINER GRAFIK
yr% = y1 + radius * rand + 5 :   REM MIT SCROLL
LINE (xl%,yl%)-(xr%,yr%),7,b
IF hoehe <> 200 GOTO sprung
  index = (yr%-yl%+1)*3*INT((xr%-xl%+16)/16)+3
  DIM a%(index)
  GET (xl%,yl%)-(xr%,yr%),a%
  PUT (xl%,yl%),a%
  PUT (xl%+50,yl%),a%,PSET
sprung:
IF breite <> 320 GOTO warten
  FOR i = 1 TO breite STEP 5
    SCROLL (1,yl%)-(breite-100,yr%),5,0
  NEXT i
warten: GOTO warten
ende: END
segment:
  CIRCLE (x1,y1),radius,f,s1*phi,s2*phi,vh
  CIRCLE (x2,y2),radius,f,s3*phi,s4*phi,vh
  CIRCLE (x3,y3),radius,f,s5*phi,s6*phi,vh
  PAINT (x,y),f
RETURN
beenden:
  WINDOW CLOSE 2 : SCREEN CLOSE 2
RETURN ende
```

Listing 2. RGB simuliert eine Farbmischung

Brücken schlagen — mit Convert 64

Wenn Sie bisher einen C 64 besessen haben, brauchen Sie auf Ihre dort angelegten Dateien nicht zu verzichten. Mit Convert 64 geben wir Ihnen ein Werkzeug, das den Datentransfer zwischen C 64 und Ihrem Amiga zum Kinderspiel macht.

Computer ist nicht gleich Computer und Zeichensatz ist nicht gleich Zeichensatz. Diese schmerzliche Erfahrung mußten schon viele Computerbesitzer machen, die Daten von einem Computer zum anderen übertragen wollten. Doch warum sollte die viele Arbeit, die man sich mit Text- und anderen Dateien gemacht hat, vergeblich gewesen sein, nur weil man auf einen neuen Computer umgestiegen ist? Diese Frage beantworten Sie spielend mit Convert 64, denn es sorgt sowohl für die Übertragung der Dateien über eine RS232-Schnittstelle, als auch für eine Umsetzung verschiedener Zeichensätze in jeden anderen beliebigen Zeichensatz. Damit nicht genug, denn Convert 64 läßt sich auch auf die Textdateien der gängigsten Textverarbeitungsprogramme für den C 64 einstellen und ist so in der Lage, direkt Texte dieser Textprogramme zu übertragen. Damit die Übertragung nicht zu lange dauert, haben sich die Programmierer noch einen ganz besonderen Trick einfallen lassen. Die Übertragung ist mit bis zu 4800 Baud möglich, obwohl das Betriebssystem des C 64 diese Geschwindigkeit eigentlich nicht vorsieht.

Geschwindigkeit ist Trumpf

Denkbar ist auch, daß Sie einen Text über einen Akustikkoppler an einen anderen Computer übermitteln. Dies hat gegenüber einem herkömmlichen Terminalprogramm den Vorteil, daß Sie Texte normaler Textprogramme verwenden können. Dabei ist es gleichgültig, ob Ihr Textprogramm seine Dateien als sequentielle- oder als Programmdatei speichert. Alle diese Probleme können Sie mit Convert 64 lösen, denn es besitzt eine universelle Konvertierungstabelle und kann auf die gängigsten Textverarbeitungssysteme eingestellt werden.

Um Convert 64 einzusetzen, benötigen Sie lediglich eine RS232-Schnittstelle bei beiden Computern, Convert 64, ein einfaches dreiadriges Kabel und natürlich einen C 64 (eventuell nochmal ausleihen). Wie das Verbindungskabel aussehen sollte, wird am Ende des Artikels beschrieben.

Geben Sie nun Convert 64 (Listing 1) mit dem MSE in Ihren C 64 ein (falls Sie nicht mehr im Besitz dieser Eingabehilfe sind, können Sie das Listing für den MSE kostenfrei beim Verlag anfordern). Verfahren Sie ebenso mit den verschiedenen Konvertierungstabellen (Listing 2 bis 7). Wenn Sie die Programme eingegeben und gespeichert haben, tippen Sie bitte

```
LOAD"CONVERT64",8
```

ein. Es dauert etwas, bis das Programm geladen worden ist. Nun starten Sie das Programm durch RUN. Es erscheint das Hauptmenü (Bild 1). In der obersten Bildschirmzeile wird immer der Name der in den Speicher geladenen Datei angezeigt. Da noch keine Datei geladen wurde, bleibt dieses Feld vorerst frei. In der sechsten Zeile steht der Name des Menüs oder der Funktion. Darunter befinden sich (bei Menüs) die Auswahlmöglichkeiten. Nach dem Programmstart befinden Sie sich immer im Hauptmenü. Von hier aus können Sie die Programmfunktionen anwählen. Zum Beispiel steht hier: »F1 Datei senden«. Das heißt nichts anderes, als daß Sie durch Drücken der Funktionstaste <F1> in das Untermenü »Datei senden« gelangen. Analog kommen Sie durch Drücken der anderen Funktionstasten in die entsprechenden Untermenüs. Falls Sie aus Versehen einmal eine falsche Taste gedrückt haben, macht dies nichts; Sie gelangen durch Drücken von <RUN/STOP> jederzeit ins übergeordnete Menü (fast immer das Hauptmenü) zurück. Es sollen nun die Menüs in ihrer Reihenfolge des Auftretens im Hauptmenü beschrieben werden. Es geht mit »Datei senden« los. Wenn Sie vom Hauptmenü aus <F1> drücken, dann gelangen Sie ins Untermenü »Datei senden«. Wenn Sie noch keine Datei geladen haben und trotzdem <F1> drücken, dann meldet das Programm »Fehler: Kein Text vorhanden!«. Solange diese Fehlermeldung dort steht, können Sie keinen Menüpunkt anwählen. Sie müssen erst irgendeine Taste drücken, dann erscheint unten auch wieder »Ihre Wahl: «. Sie müssen also zunächst den Menüpunkt »Datei laden« anwählen.

Wenn Sie vom Hauptmenü <F3> drücken, dann gelangen Sie ins Untermenü »Datei laden« (Bild 2). Hier wird der Name der Datei eingegeben, die Übertragen werden soll. Wenn Sie alles richtig eingestellt haben, dann drücken Sie <F1>. Geben Sie nun den Namen der Datei ein. Wenn Sie das Programm auf »aus Speicher senden« (<F5>) gestellt haben, dann wird die Datei sofort in den Speicher geladen. Falls hierbei ein Fehler auftritt, wird die Fehlermeldung von der Diskettenstation geholt und in der obersten Bildschirmzeile ausgegeben. Das Programm wartet dann auf einen Tastendruck.

Konvertierung von Text-Dateien

Bitte beachten Sie, daß das Laden in den Speicher nur bei Dateien mit einer Länge bis zu 175 Blöcken möglich ist, andernfalls stellen Sie das Programm auf »von Diskette senden« (mit <F5>).

Dies ist nur möglich, wenn die Dateiart auf ASCII steht. Wenn Dateien von Textprogrammen übertragen werden sollen, müssen diese erst in den Speicher geladen werden. Mit <F3> können Sie die Art (Quelle) der Datei einstellen. Durch Drücken von <F3> wird immer um eine Möglichkeit weitergeschaltet. Es gibt folgende Möglichkeiten:

Star-Texter — die Datei stammt von dem Textverarbeitungsprogramm Star-Texter.

Vizawrite — eine Textdatei von Vizawrite.

ASCII — eine beliebige Datei, hier sind Sie auf kein bestimmtes (Textverarbeitungs-)Programm festgelegt, (mit dieser Einstellung können zum Beispiel Basic-Programme oder Adreßdateien übertragen werden).



Bild 1. Das Hauptmenü, von hier aus erreichen Sie alle Funktionen des Übertragungsprogramms »Convert 64«

Textomat — Die Textdatei stammt von Texomat.
Textomat+ — eine Textdatei von Textomat+.
Master Text — eine Master Text-Datei soll übertragen werden.

Die Funktionstaste <F5> hat eine Schalterfunktion. Mit ihr kann man wählen, ob der Text vor dem Senden in den Speicher geladen und auch von dort gesendet oder ob direkt von Diskette übertragen wird (nur möglich, wenn die Dateiart auf ASCII gestellt worden ist). Nachteilig ist an der zweiten Methode, daß sie nur geringe Übertragungsgeschwindigkeiten erlaubt (weniger als 600 Baud). Allerdings unterliegen die zu übertragenden Dateien keiner Beschränkung in ihrer Länge. Dateien mit bis zu 175 Blöcken können in den Speicher geladen und direkt von dort auch mit bis zu 4800 Baud gesendet werden, so daß »senden direkt von Diskette« wohl nur in den seltensten Fällen genutzt wird. Mit <F7> kann man den Dateityp nur dann verstellen, wenn als Dateiart auch ASCII gewählt worden ist. Andernfalls steht der Dateityp auf PRG (Programmdatei). Das liegt daran, daß alle Textverarbeitungsprogramme für den C 64 die Texte als Programmdatei ablegen. In der Dateiart ASCII läßt sich der Dateityp zwischen PRG und SEQ (Sequentielle Datei) umschalten. Der Dateityp muß der Kennzeichnung im Inhaltsverzeichnis entsprechen. Wenn Sie Dateien des Textprogrammes Star-Texter übertragen wollen, dann denken Sie bitte daran, daß die Absätze markiert sind, andernfalls werden unnötige Leerzeichen übertragen. Textdateien anderer Textprogramme können selbstverständlich auch übertragen werden. Stellen Sie die Dateiart dazu auf ASCII und den Dateityp auf den Typ der Texte, das heißt im Inhaltsverzeichnis steht zum Beispiel:

```
20 » Text Beispiel « SEQ
```

dann stellen die den Dateityp auf SEQ.

Nehmen wir also an, daß Sie einen Text in den Speicher geladen haben oder direkt von Diskette senden wollen, dann können Sie im Menü »Datei senden« (Bild 3) wieder auswählen. Bitte beachten Sie, daß sich die Bedeutung der Funktionstasten gegenüber der im Hauptmenü unterscheidet! Wenn Sie nun <F1> drücken, wird die im Speicher oder auf Diskette befindliche Datei über die serielle Schnittstelle übertragen. (Bedenken Sie, daß beide Computer auf die gleichen Übertragungsparameter (Baudrate, Stopbits usw.) eingestellt sein müssen, damit auch ein (sinnvoller) Text empfangen wird). Mit <F3> können Sie einstellen, ob am Anfang und Ende ein STX bzw.

Text und Steuerzeichen senden

ETX gesendet wird. Die Funktionstaste dient hier sozusagen als Schalter. Sie können hier zwischen »ja« und »nein« umschalten. Falls hinter dem Menüpunkt ein »ja« steht, wird die entsprechende Funktion auch ausgeführt. Wenn hinter STX/ETX also »ja« steht, dann wird am Textanfang ein STX (ASCII-Wert 2) und am Textende ein ETX (ASCII-Wert 3) gesendet. Dies macht selbstverständlich auch nur einen Sinn, wenn das Terminalprogramm für den Amiga mit diesen Zeichen etwas anzufangen weiß. STX (Start of Text) kennzeichnet den Textanfang, ETX (End of Text) das Textende. Diese Steuerzeichen werden dazu benutzt, dem Amiga mitzuteilen, wann er anfangen muß, den Text zu speichern und wann er aufhören muß. Aber wie gesagt, nur wenn der andere Computer diese Option auch anbietet, dann ist sie sinnvoll einzusetzen. Da je-

Anpassung an Amiga-Terminalprogramme

doch fast alle Terminalprogramme für den Amiga diese Funktion beherrschen, sollten Sie versuchsweise mit diesen Möglichkeiten experimentieren, um optimale Ergebnisse zu erzielen.

```
Name:                           Convert 64

 Datei laden
  f1   Datei laden
  f3   Dateiart einstellen    (Star-Texter)
  f5   von Diskette senden
  f7   Dateityp  seq
r/s    Zurueck

Ihre Wahl: █
```

Bild 2. Im Menü »Datei laden« können Sie viele Textprogramme auswählen, um deren Dateien zu konvertieren



Bild 3. Die Dateien können mit verschiedenen Parametern und Steuerzeichen (etwa LF nach CR) übermittelt werden

Nun zum dritten Menüpunkt (»EOT«). Auch hier wirkt die Funktionstaste wieder als Schalter. Falls dieser Menüpunkt auf ja gesetzt ist, dann wird am Textende ein EOT (ASCII-Wert 4, End of Transmission) übertragen. Wenn das Terminalprogramm für den Amiga über die Option verfügt, dann wird automatisch die Übertragung beendet. Mit der als Schalter wirkenden Funktionstaste <F7> wird die Benutzung der Konvertierungstabelle erlaubt (ja) oder »verboten« (nein). Diese Funktion braucht man hauptsächlich, wenn man statt einer Textdatei (Einstellung »ja«) ein sogenanntes Binärfile (Einstellung »nein«), also ein Maschinensprache- oder Basic-Programm übertragen möchte. Die Konvertierungstabelle ist für eine einwandfreie Übertragung von Texten unerläßlich, bei Binärdateien würde die Tabelle nur zu Fehlern führen. Mit der Funktionstaste <F2> läßt sich ein beliebiges Zeichen unterdrücken. Im Normalfall ist diese Option ausgeschaltet. Sie kann durch Drücken von <F2> aktiviert werden. Dann erscheint hinter dem Menüpunkt ein Cursor. Sie können hier dann den Wert des Zeichens eingeben, das unterdrückt werden soll. Falls es sich um ein darstellbares Zeichen (also kein Steuerzeichen) handelt, wird es zusätzlich noch angezeigt. Zur Eingabe: Sie können hier nur Ziffern eingeben, andere Zeichen werden von vornherein ignoriert. Als Editiermöglichkeiten stehen hier zur Verfügung:

<HOME> — gehe an den Anfang der Eingabe.
<CLR> — löscht die gesamte Eingabe.
<RUN/STOP> — Abbruch der Eingabe, die Option wird ausgeschaltet und es wird wieder ins Untermenü Datei senden zurückgekehrt.
Cursor-Tasten — (erlaubt ist hier nur Cursor rechts/Cursor links) bewegen den Cursor.
<SPACE> — überschreibt das Zeichen, auf dem der Cursor steht.
<DEL> — löscht das Zeichen links vom Cursor und zieht den Rest der Eingabe nach.
<INS> — schafft Platz für ein Zeichen (Ziffer) an der Cursorposition, der Rest der Eingabe wird nach rechts geschoben und eventuell fallen am Ende auch Zeichen heraus.
<RETURN> — Eingabe ordnungsgemäß abschließen und Eingabe übernehmen (im Ge-



Bild 4. Das Parametermenü ist zum Einstellen der richtigen Werte für die Übertragung der Dateien wichtig

gensatz zu <RUN/STOP>). Im großen sind die Tastenfunktionen gleich denen des Basic-Editors. So löscht zum Beispiel <CLR> im Basic-Editor den ganzen Bildschirm, in Convert 64 nur das Eingabefeld. Wenn diese Option eingeschaltet ist (es steht dort also eine Zahl und eventuell auch ein Zeichen), dann wird bei der Übertragung folgendermaßen verfahren:

1. Zeichen holen (egal, ob von Floppy-Laufwerk oder Speicher).
2. Ist eine Konvertierung des Zeichens gewünscht (also Konvertierungstabelle eingeschaltet?).
3. Falls ja: Zeichen konvertieren.
4. Zeichen unterdrücken (also Option »Zeichen unterdrükken« eingeschaltet)?
5. Falls ja: Ist auszugebendes Zeichen = zu unterdrückendes Zeichen?
6. Falls ja: gehe zu Schritt 1.
7. Nein: Gib das Zeichen aus.

Zusatzfunktionen zur Konvertierung

Man sieht schon, daß bei der Benutzung der Konvertierungstabelle mehr als ein Zeichen unterdrückt werden kann, indem man in der Konvertierungstabelle für die zu unterdrückenden Zeichen einen beliebigen, aber für alle gleichen Wert einträgt und hier dann den Wert für alle Zeichen angibt und die Konvertierungstabelle einschaltet. Wenn gewünscht, kann man nach jedem Return (ASCII-Wert 13) noch ein Line-Feed (ASCII-Wert 10) senden lassen. Auch hier dient die Funktionstaste <F4> als Schalter. Wenn diese Option eingeschaltet ist, dann wird nach jedem auszugebenden Return noch ein Line-Feed gesendet. Hiermit hat es folgende Bewandtnis: Wenn man einen Text überträgt, wird nach jeder vollständig übertragenen Zeile noch ein Zeilenendekennzeichen mitgeschickt.

Sie können sich jederzeit das Inhaltsverzeichnis der im Laufwerk befindlichen Diskette ansehen, es genügt, wenn Sie im Hauptmenü auf <F5> drücken. Der Bildschirm wird dann gelöscht und das Inhaltsverzeichnis wird ausgegeben. Wenn Sie während der Ausgabe <SHIFT> drücken, wird die Ausgabe so lange gestoppt, wie Sie die Taste gedrückt halten. Fehlermeldungen werden wie üblich angezeigt, bei dieser Funktion wird der Fehlerkanal aber immer gelesen, auch wenn kein Fehler auftritt. Durch Drücken einer beliebigen Taste kehren Sie dann wieder in das Hauptmenü zurück.

Mit <F7> gelangen Sie in das Parameter-Untermenü. Hier haben Sie die in Bild 4 gezeigten Wahlmöglichkeiten. Wenn Sie nun <RUN/STOP> drücken, gelangen Sie zurück in das Hauptmenü. Die RS232-Einstellung können Sie durch Drücken der Taste <F1> aufrufen. Mit <F1> können Sie die Baudrate verstellen. Die Baudrate gibt an, wie viele Bits pro Sekunde (maximal) übertragen werden. Es gibt hier folgende verschiedene Baudraten: 50, 75, 110, 134.5, 150, 200, 300, 600, 1200, 1800, 2000, 2400, 3600 und 4800 Baud.

Damit überhaupt eine Datei übertragen werden kann, müssen beide Computer auf die gleichen Baudraten gestellt werden. In Ihrem Interesse sollten Sie eine möglichst hohe Baudrate nehmen, da sich dann die Übertragungszeit stark verkürzt. Sie sollten allerdings auch an die Übertragungssicherheit denken. Je höher die Baudrate und je länger das die beiden Computer verbindenden Kabel, desto mehr Fehler treten bei der Übertragung auf. Die Anzahl der Datenbits können Sie durch Drücken von <F3> ändern. Mögliche Werte sind hier 6, 7 und 8 Datenbits. Mit <F5> können Sie die Anzahl der Stopbits wählen: 1 oder 2 Stopbits. Die Parität läßt sich schließlich mit <F7> einstellen (keine, gerade, ungerade, 8. Datenbit = 0 und 8. Datenbit = 1). Mit <RUN/STOP> gelangen Sie in das übergeordnete Menü.

Wenn Sie im Parametermenü <F3> drücken, dann wird die Zeichentabelle (Konvertierungstabelle) ausgegeben. Durch Drücken von <RUN/STOP> gelangen Sie in das Parametermenü zurück. Die Zeichentabelle (Bild 5) dient zur Anpassung der Zeichen an einen anderen Computer. Dies ist (leider) notwendig, da es noch keinen Standard-Code für alle Computer gibt. ASCII hat sich zwar weitgehend durchgesetzt, aber zum Beispiel Commodore und IBM kochen ohnehin ihre eigenen »Süppchen«. So sind zum Beispiel im Commodore-Zeichensatz gegenüber dem ASCII-Wert die Groß- und Kleinbuchstaben vertauscht. Die Konvertierungstabelle besteht aus vier Spalten: zwei numerischen und zwei »Text«-Spalten. Wenn die Konvertierungstabelle bei der Übertragung eingeschaltet ist (vergleiche »Datei senden«), dann wird folgendermaßen verfahren:

1) Zeichen aus Speicher oder von Diskette holen.
2) »Nachschlagen«, was dem Zeichen für ein anderer Wert zugeordnet ist.
3) Zeichen übertragen.

Beispiel: Wenn sie im ganzen Text das große »A« durch das kleine »a« ersetzen wollen, dann suchen Sie (mit Cursorhoch/Cursor-runter die Stelle anfahren) in der Tabelle die entsprechende Stelle auf (Code von »A«: 97). Dort steht dann: »97 97 A A«. Sie können nur die zweite Zahl dort ändern. Diese gibt an, was für ein Wert gesendet wird, wenn als (Orginal-)Wert 97 in der Datei vorkommt. Hinter den Zahlen stehen dann die entsprechenden Zeichen, wenn die Zahlen keine Steuerzeichen sind. Sie ändern nun diese Zahl in 65 (dies entspricht dem »a«). Löschen Sie den alten Wert durch Drücken von <CLR> (sie können den alten Wert auch mit der Zahl überschreiben und den Rest der alten Zahl durch Betätigen der Leertaste überschreiben) und tippen Sie die Zahl.

Wenn sie <RETURN> drücken, dann wird der alte Wert durch den neuen ersetzt und das Zeichen entsprechend geändert. Außerdem ändert sich das letzte Zeichen.

Viele Editiermöglichkeiten

Alle großen »A« werden jetzt durch kleine »a« ersetzt (diese Formulierung ist eigentlich unpräzise, man sollte besser sagen, daß Zeichen, die den Wert 97 haben, durch Zeichen mit dem Wert 65 ersetzt werden). Dies ist notwendig, da nicht alle Computer, wie schon er-

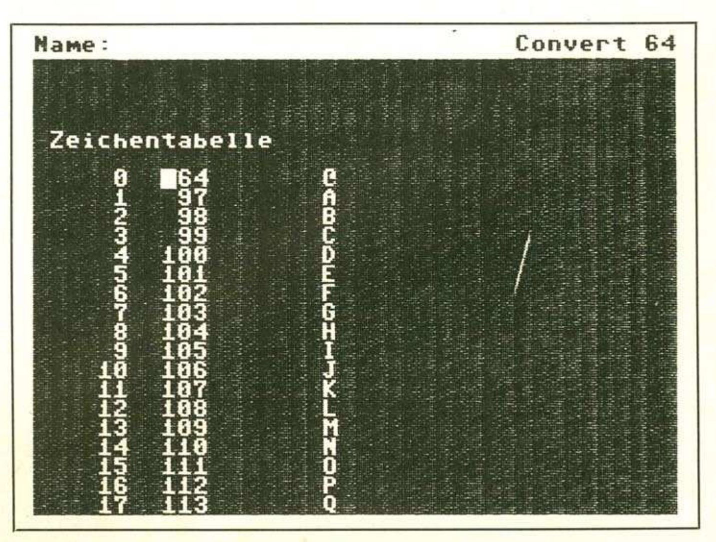


Bild 5. Sie können praktisch jedes Zeichen für jedes Zeichen senden, um so ASCII-Unterschiede auszugleichen

wähnt, die gleichen Zeichensätze haben.

Und nun zu den komfortablen Editiermöglichkeiten. Neben den bereits beschriebenen Möglichkeiten kommen noch folgende hinzu:
<CRSR DOWN> — Cursor eine Zeile tiefer, am unteren Rand wird die Tabelle dann gescrollt.
<CRSR UP> — Cursor eine Zeile höher, am oberen Rand wird die (<SHIFT+CRSR DOWN>) Tabelle gescrollt.
<F5> — 32 Zeichen hoch.
<F7> — 32 Zeichen runter.
<F2> — Position für Tabulator 1 merken.
<F4> — Position für Tabulator 2 merken.
<F3> — gemerkte Position für Tabulator 2 anspringen.
<RUN/STOP> — zurück in das Parametermenü.

Die Listings 2 bis 7 sind Beispiel-Konvertierungstabellen. Das Listing mit dem Namen »Standard« wird beim Starten von Convert 64 automatisch geladen und bestimmt die Grundeinstellung der Konvertierungstabelle und der Übertragungsparameter, so lange, bis Sie eine neue Tabelle laden.

Konvertierungstabellen für Text-Dateien

Die Tabellen können Sie laden, wenn Sie vom Parametermenü aus <F5> »Tabellen laden« drücken. Geben Sie nun den Namen der Datei ein, und drücken Sie <RETURN>. Der Computer sucht nun die Datei auf Diskette und lädt sie. Falls ein Diskettenfehler auftritt, wird er in der obersten Zeile angezeigt und der Computer wartet auf einen Tastendruck. Bitte beachten Sie, daß die Konvertierungstabelle wie auch die RS232-Einstellung dadurch überschrieben werden. Sichern Sie also erst ihre angelegten Tabellen auf Diskette. Die Tabellen können Sie durch Drücken von <F7> speichern. Die Eingabe verläuft analog zu »Tabellen laden«. Hier werden lediglich die Tabellen (Konvertierungstabelle und RS232-Einstellung) auf Diskette gespeichert.

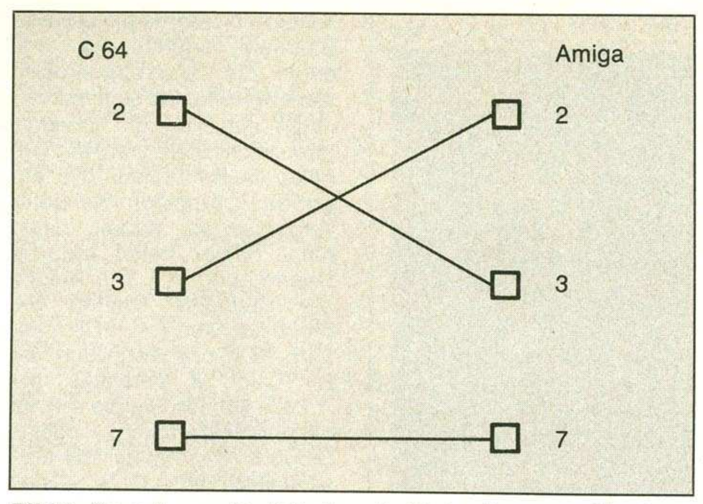


Bild 6. So müssen die Kabel verdrahtet sein, wenn Sie den C 64 an den Amiga anschließen

Wenn Sie vom Hauptmenü aus <F2> drücken, dann gelangen Sie in das Untermenü »Farben«. Sie können hier Rahmen-, Hintergrund- und Schriftfarben wählen. Durch Drücken von <RUN/STOP> gelangen Sie wieder in das Hauptmenü zurück. Zum Menü Diskbefehle wird man wohl kaum noch etwas sagen müßen. Sie haben für die Befehlseingabe (S,V,I,N) wieder alle Editiermöglichkeiten wie gewohnt zur Verfügung. Drücken Sie dann irgendeine Taste, um ins Hauptmenü zurückzugelangen. (Während der Eingabe können Sie auch durch Drücken von <RUN/STOP> zurück ins Hauptmenü.) Das Programm können Sie durch Drücken von <F6> verlassen. Achtung: Es erfolgt keine Sicherheitsabfrage. Dies ist aber kein Nachteil, da Sie das Programm jederzeit neu starten können (»RUN« und <RETURN> normalerweise, beziehungsweise wenn Sie versehentlich einen Reset ausgelöst haben, dann geben Sie »SYS 2144« ein und drücken <RETURN>). Alle Einstellungen sind noch wie zuvor, und ein geladener Text ist noch im Speicher vorhanden.

Vom C 64 zum Amiga

Als Beispiel wollen wir Ihnen zeigen, wie Sie eine Vizawrite 64-Textdatei auf den Amiga übertragen können. Dazu müssen Sie sich zunächst ein Verbindungskabel wie in Bild 6 gezeigt löten. Besorgen Sie sich für den C 64 eine RS232-Schnittstelle, wie sie für Akustikkoppler erhältlich ist und verbinden Sie beide Computer. Dann starten Sie Convert 64, drücken <F7> (Parameter) und dann <F5> (Tabellen laden). Geben Sie nun den Namen der Konvertierungstabelle (in diesem Fall Vizawrite) ein — die Übersetzungstabelle für Vizawrite wird jetzt geladen. Drücken Sie nun <F1> <1> (RS232-Einstellung) und stellen Sie Ihr Terminalprogramm auf dem Amiga auf die gleichen Werte ein. Durch zweimaliges Drücken auf <Run/Stop> gelangen Sie wieder in das Hauptmenü. Hier drücken Sie <F3> (Datei laden) und stellen dort mit <F3> das Textprogramm »Vizawrite 64« ein.

Drücken Sie nun auf <F1> und geben den genauen Namen des Textes ein — die Datei wird nun geladen und der Name erscheint in der obersten Zeile. Sie befinden sich nun automatisch im Menü »Datei senden«. Jetzt brauchen Sie nur noch die Konvertierungstabelle mit <F7> einzuschalten und die Datenübertragung mit <F1> zu starten. Ihr Text wird dann auf den Amiga übertragen. Bei anderen Textdateien brauchen Sie nur die entsprechenden Parameter einstellen, ebenso bei anderen Computern. (Martin Müller/aw/dm)

Der MSE ist eine Eingabehilfe für den C 64, der die Fehlerquote (Tippfehler) bei der Eingabe von Maschinensprache-Programmen vermindert. Sollten Sie nicht über den MSE verfügen, können Sie eine Kopie des MSE-Listings bei der 64'er- oder Amiga-Redaktion anfordern. Legen Sie dazu bitte einen frankierten Rückumschlag mit Ihrer Adresse und dem Vermerk »MSE« bei.

```
Name : convert 64           0801 1e81
-------------------------------------
0801 : 4a 08 ff ff 9e 20 32 31   65
0809 : 34 34 3a 8f 20 20 43 4f   87
0811 : 4e 56 45 52 54 20 36 34   ae
0819 : 20 20 20 20 20 20 20 20   19
0821 : 20 20 20 20 20 20 20 20   21
0829 : 20 20 20 20 20 20 20 20   29
0831 : 20 20 20 20 20 20 20 20   31
0839 : 20 20 20 20 20 20 20 20   39
0841 : 20 20 20 20 20 20 20 20   41
0849 : 00 00 00 00 00 00 00 00   4a
0851 : 00 00 00 00 00 00 00 00   52
0859 : 00 00 00 00 00 00 00 a9   ad
0861 : 0e 20 d2 ff a9 08 20 d2   35
0869 : ff a9 00 8d 21 d0 8d 20   fe
0871 : d0 20 59 0c 20 1c 0c a2   82
0879 : 05 a0 16 20 e7 0b 20 15   d9
0881 : 0c 20 e4 ff c9 85 f0 22   a7
0889 : c9 86 f0 21 c9 87 d0 09   24
0891 : 20 3b 0d 20 37 0c 4c 75   86
0899 : 08 c9 88 f0 13 c9 89 f0   4d
08a1 : 15 c9 8a f0 0e c9 8b d0   5a
08a9 : d8 60 4c a5 10 4c 7e 09   e9
08b1 : 4c f2 08 4c d6 1b 20 1c   01
08b9 : 0c a2 a3 a0 16 20 e7 0b   2b
08c1 : 20 15 0c 20 e4 ff c9 03   ee
08c9 : f0 aa c9 85 d0 06 ee 20   6b
08d1 : d0 4c c4 08 c9 86 d0 06   1a
08d9 : ee 21 d0 4c c4 08 c9 87   d8
08e1 : d0 e1 ee 86 02 4c b7 08   a0
08e9 : 4c 47 19 4c 02 1b 4c 8c   ec
08f1 : 1a 20 1c 0c a2 3c a0 17   61
08f9 : 20 e7 0b 20 15 0c 20 e4   d0
0901 : ff c9 03 d0 03 4c 75 08   38
0909 : c9 86 f0 dc c9 87 f0 db   41
0911 : c9 88 f0 da c9 85 d0 e6   90
0919 : 20 1c 0c a2 b3 a0 17 20   7c
0921 : e7 0b 20 15 0c 20 f4 0a   e2
0929 : 20 e4 ff c9 03 d0 03 4c   50
0931 : f2 08 c9 85 d0 10 ae 30   f3
0939 : 1e e8 e0 0e 90 02 a2 00   69
0941 : 8e 30 1e 4c 26 09 c9 86   d7
0949 : d0 10 ae 31 1e e8 e0 03   a6
0951 : 90 02 a2 00 8e 31 1e 4c   0e
0959 : 26 09 c9 87 d0 0b ad 32   e8
0961 : 1e 49 01 8d 32 1e 4c 26   a7
0969 : 09 c9 88 d0 bb ae 33 1e   cd
0971 : e8 e0 05 90 02 a2 00 8e   6f
0979 : 33 1e 4c 26 09 20 1c 0c   ad
0981 : a2 7b a0 18 20 e7 0b 20   ba
0989 : 15 0c 20 53 0b 20 e4 ff   5c
0991 : c9 03 d0 03 4c 75 08 c9   94
0999 : 86 d0 1c ae 34 1e e8 e0   fe
09a1 : 06 90 02 a2 00 8e 34 1e   46
09a9 : e0 02 f0 de a9 00 8d 3a   e7
09b1 : 1e 8d 3f 1e 4c 8b 09 c9   02
09b9 : 87 d0 12 ad 34 1e c9 02   42
09c1 : d0 c8 ad 3a 1e 49 01 8d   f3
09c9 : 3a 1e 4c 8b 09 c9 88 d0   39
09d1 : 13 ad 34 1e c9 02 d0 b5   e7
09d9 : ad 3f 1e 49 01 8d 3f 1e   8c
09e1 : 4c 8b 09 ea c9 85 d0 a5   ea
09e9 : a2 40 a0 1e 20 7b 0b a2   e7
09f1 : 40 a0 1e 86 f7 84 f8 20   a1
09f9 : 47 0c 10 03 4c 75 08 a2   81
0a01 : 00 bd 40 1e 20 35 10 9d   db
0a09 : 50 1e bd 40 1e 09 80 9d   47
0a11 : 40 1e e8 e0 10 90 ea a9   3b
0a19 : 50 a2 1e a0 0f 20 e2 0a   e8
0a21 : f0 c6 ad 34 1e a2 0f c9   2d
0a29 : 03 90 10 f0 07 c9 05 f0   4b
0a31 : 03 a9 d4 2c a9 54 9d 50   18
0a39 : 1e 86 02 ad 3a 1e d0 3b   1f
0a41 : ad 3f 1e f0 03 4c be 1c   f9
0a49 : a9 01 a2 08 a0 00 20 ba   1c
0a51 : ff a6 02 e8 8a a2 50 a0   81
```

```
0a59 : 1e 20 bd ff a9 00 a2 80  1d
0a61 : a0 20 20 d5 ff 90 0e a9  e4
0a69 : 00 8d 36 1e 20 65 0d 20  23
0a71 : 37 0c 4c 7e 09 8e 35 1e  a7
0a79 : 8c 36 1e 4c a5 10 a9 20  f3
0a81 : 4c 19 0d 20 7f 0a 4c 19  4d
0a89 : 0d 20 84 0a a2 37 a0 1e  ab
0a91 : 86 f7 84 f8 a9 03 8d 39  ae
0a99 : 1e a9 01 a2 03 a4 d6 20  11
0aa1 : 4a 10 20 84 0a 20 b7 1b  43
0aa9 : a9 03 8d 39 1e a9 01 a2  d7
0ab1 : 08 a4 d6 20 4a 10 20 84  74
0ab9 : 0a 20 84 0a ad 37 1e 20  83
0ac1 : e9 0f 90 02 a9 20 20 19  e5
0ac9 : 0d 20 84 0a ae 37 1e bd  e1
0ad1 : 30 1d 20 54 1b b0 04 20  aa
0ad9 : e9 0f 2c a9 20 20 19 0d  0c
0ae1 : 60 85 f9 86 fa b1 f9 c9  0c
0ae9 : 20 d0 03 88 10 f7 84 02  1a
0af1 : c0 ff 60 a2 0f a0 07 20  70
0af9 : a0 0c a9 00 ae 30 1e f0  d0
0b01 : 06 18 69 06 ca d0 fa aa  a3
0b09 : a9 06 85 02 bd e8 16 20  13
0b11 : 19 0d e8 c6 02 d0 f5 a2  87
0b19 : 12 a0 09 20 a0 0c ad 31  45
0b21 : 1e 18 69 36 20 19 0d a2  b1
0b29 : 12 a0 0b 20 a0 0c ad 32  d8
0b31 : 1e 18 69 31 20 19 0d a2  20
0b39 : 0f a0 0d 20 a0 0c ac 33  63
0b41 : 1e be 76 18 bd 26 18 c9  60
0b49 : ff f0 06 20 19 0d e8 d0  85
0b51 : f3 60 a2 1b a0 09 20 a0  94
0b59 : 0c ac 34 1e be 2e 19 bd  c9
0b61 : da 18 c9 ff f0 06 20 19  ac
0b69 : 0d e8 d0 f3 a2 05 a0 0b  88
0b71 : 20 a0 0c 20 83 1c 20 a0  c3
0b79 : 1c 60 86 f9 84 fa a2 34  b9
0b81 : a0 19 20 e7 0b a9 20 20  72
0b89 : 19 0d a0 00 84 02 a4 02  40
0b91 : b1 f9 29 7f 20 19 0d e6  46
0b99 : 02 a5 02 c9 10 d0 ef 60  30
0ba1 : 20 d6 0c a2 05 bd ff 15  ec
0ba9 : 20 19 0d ca 10 f7 a2 00  3e
0bb1 : bd 40 1e 20 19 0d e8 e0  79
0bb9 : 10 90 f5 a2 08 a9 a0 20  74
0bc1 : d5 0b a2 09 bd 65 1e 20  a5
0bc9 : 19 0d ca 10 f7 20 dd 0c  2d
0bd1 : a2 28 a9 77 20 19 0d ca  75
0bd9 : d0 fa 60 a0 00 b1 61 e6  33
0be1 : 61 d0 02 e6 62 60 86 61  0e
0be9 : 84 62 20 dc 0b c9 ff f0  23
0bf1 : 22 c9 fd d0 0e 20 dc 0b  fd
0bf9 : aa 20 dc 0b a8 20 a0 0c  72
0c01 : 4c eb 0b c9 fe d0 06 20  0e
0c09 : dd 0c 4c eb 0b 20 19 0d  ad
0c11 : 4c eb 0b 60 a2 93 a0 16  97
0c19 : 4c e7 0b 20 ee 0c 20 a1  33
0c21 : 0b a2 00 a0 02 20 a0 0c  4d
0c29 : a2 00 bd 3f 1c 20 19 0d  64
0c31 : e8 e0 1e d0 f5 60 20 e4  d8
0c39 : ff c9 00 f0 f9 60 a2 3e  e5
0c41 : a0 03 86 f7 84 f8 a9 10  da
0c49 : 8d 39 1e a9 00 a2 10 a0  c6
0c51 : 16 20 5e 0e ad 3c 1e 60  c7
0c59 : a9 00 20 90 ff 20 1c 0c  a6
0c61 : a2 8c a0 0c 20 e7 0b a9  b4
0c69 : 01 a2 08 a0 00 20 ba ff  bd
0c71 : a9 08 a2 84 a0 0c 20 bd  be
0c79 : ff a9 00 a2 30 a0 1d 20  5e
0c81 : d5 ff 60 53 54 41 4e 44  e9
0c89 : 41 52 44 fd 01 0a 4d 0f  78
0c91 : 0d 05 0e 14 20 2e 2e 2e  b0
0c99 : ff 00 00 00 00 00 00 86  a6
0ca1 : d3 84 d6 98 0a 0a 0a 85  a3
0ca9 : 71 85 28 a2 00 86 72 86  46
0cb1 : 29 26 28 26 29 26 28 26  6d
0cb9 : 29 18 a5 71 65 28 85 d1  d7
0cc1 : 85 f3 a5 72 65 29 85 29  ff
0cc9 : 18 69 d8 85 f4 a5 29 6d  79
0cd1 : 88 02 85 d2 60 a2 00 a0  72
0cd9 : 00 4c a0 0c a9 00 85 d3  01
0ce1 : a6 d3 a4 d6 c0 18 b0 04  0c
0ce9 : c8 4c a0 0c 60 20 d6 0c  fb
0cf1 : a2 03 a0 00 a9 20 91 d1  c2
0cf9 : ad 86 02 91 f3 c8 d0 f4  4f
0d01 : e6 d2 e6 f4 ca d0 ed a9  e7
0d09 : 20 91 d1 ad 86 02 91 f3  c2
0d11 : c8 c0 e8 d0 f2 4c d6 0c  92
0d19 : 48 a4 d3 c0 28 90 03 20  14
0d21 : dd 0c a4 d3 ad 86 02 91  e2
0d29 : f3 68 91 d1 e6 d3 60 a6  cb
0d31 : d3 a4 d6 f0 04 88 20 a0  70
0d39 : 0c 60 a9 93 20 d2 ff a2  30
0d41 : 00 bd 4a 0e 20 d2 ff e8  df
0d49 : e0 13 90 f5 a9 01 a2 08  d3
0d51 : a0 00 20 ba ff a9 01 a2  e7
0d59 : 5d a0 0e 20 bd ff 20 c0  6c
0d61 : ff 4c b9 0d 20 cc ff a9  52
0d69 : 01 20 c3 ff a9 0f a2 08  19
0d71 : a0 0f 20 ba ff a9 00 20  86
0d79 : bd ff 20 c0 ff a2 0f 20  e7
0d81 : c6 ff 20 d6 0c a9 a0 a2  00
0d89 : 28 20 19 0d ca d0 fa 20  09
0d91 : dd 0c a9 77 a2 28 20 19  ec
0d99 : 0d ca d0 fa 20 d6 0c 20  c8
0da1 : cf ff a6 90 d0 08 09 80  9e
0da9 : 20 19 0d 4c a0 0d 20 cc  af
0db1 : ff a9 0f 20 c3 ff 38 60  2b
0db9 : a2 01 20 c6 ff 20 35 0e  af
0dc1 : 20 35 0e 20 22 0e a9 20  7d
0dc9 : 20 d2 ff 20 ec 0d 20 35  78
0dd1 : 0e 20 22 0e 20 e2 0d ad  e2
0dd9 : 8d 02 29 01 d0 f9 4c cf  7f
0de1 : 0d 38 20 f0 ff a0 04 18  76
0de9 : 20 f0 ff a9 00 85 f8 20  07
0df1 : cf ff f0 2c a6 f8 d0 08  07
0df9 : c9 20 f0 f3 a2 01 86 f8  cb
0e01 : c9 12 d0 07 a9 01 85 c7  31
0e09 : 4c f0 0d c9 92 d0 07 a9  69
0e11 : 00 85 c7 4c f0 0d a2 01  53
0e19 : 86 d4 20 d2 ff 4c f0 0d  ac
0e21 : 60 a9 0d 20 d2 ff 20 cf  ea
0e29 : ff 85 f7 20 cf ff a6 f7  74
0e31 : 20 cd bd 60 20 cf ff 20  74
0e39 : cf ff a5 90 d0 01 60 68  eb
0e41 : 68 a9 0d 20 d2 ff 4c 65  ee
0e49 : 0d c9 4e 48 41 4c 54 53  46
0e51 : 56 45 52 5a 45 49 43 48  66
0e59 : 4e 49 53 0d 24 8d 3b 1e  9a
0e61 : 20 a0 0c 18 a5 d1 65 d3  fe
0e69 : 85 d1 a5 d2 69 00 85 d2  ed
0e71 : a9 00 85 d3 20 e0 0f 20  7c
0e79 : e4 ff c9 00 f0 f9 c9 03  db
0e81 : d0 09 a2 80 20 e0 0f 8e  f1
0e89 : 3c 1e 60 c9 1d f0 65 c9  a8
0e91 : 9d f0 52 c9 13 f0 6f c9  7e
0e99 : 93 f0 75 c9 91 f0 40 c9  70
0ea1 : 11 f0 3f c9 0d f0 35 c9  f4
0ea9 : 14 f0 2e c9 94 f0 72 ae  f2
0eb1 : 3b 1e 10 0f c9 85 90 0b  02
0eb9 : c9 8b b0 07 38 e9 82 aa  87
0ec1 : 4c 85 0e 20 1c 10 b0 af  bc
0ec9 : a4 d3 cc 39 1e d0 03 4c  be
0ed1 : 78 0e 91 d1 e6 d3 4c 75  18
0ed9 : 0e 4c 4d 0f 4c 7b 0f 4c  b8
0ee1 : c1 0f 4c d2 0f a5 d3 d0  a6
0ee9 : 03 4c 78 0e 20 e0 0f c6  c5
0ef1 : d3 4c 75 0e a4 d3 cc 39  98
0ef9 : 1e d0 03 4c 78 0e 20 e0  04
0f01 : 0f e6 d3 4c 75 0e 20 e0  0c
0f09 : 0f a0 00 84 d3 4c 75 0e  8b
0f11 : 20 e0 0f ac 39 1e 88 a9  f5
0f19 : 20 91 d1 88 10 fb 4c 0a  ad
0f21 : 0f ac 39 1e c4 d3 d0 03  cd
0f29 : 4c 78 0e 20 e0 0f ac 39  e5
0f31 : 1e 88 c4 d3 f0 0f ac 39  eb
0f39 : 1e 88 88 b1 d1 c8 91 d1  41
0f41 : 88 c4 d3 d0 f5 a9 20 91  8a
0f49 : d1 4c 75 0e 20 e0 0f a4  ee
0f51 : d3 d0 02 f0 22 cc 39 1e  d5
0f59 : d0 06 ac 39 1e 4c 6d 0f  97
0f61 : b1 d1 88 91 d1 c8 c8 cc  6f
0f69 : 39 1e d0 f4 88 a9 20 91  fd
0f71 : d1 a4 d3 f0 02 c6 d3 4c  e5
0f79 : 75 0e a9 00 8d 3c 1e 20  d3
0f81 : e0 0f ad 3b 1e 29 01 d0  8c
0f89 : 0d ac 39 1e 88 b1 d1 91  7f
0f91 : f7 c0 00 d0 f7 60 ac 39  aa
0f99 : 1e a9 00 99 3e 03 88 b1  41
0fa1 : d1 99 3e 03 c0 00 d0 f6  6c
0fa9 : a2 3e a0 03 86 7a 84 7b  38
0fb1 : 20 79 00 20 f3 bc 20 9b  6f
0fb9 : bc a0 00 a5 65 91 f7 60  fe
0fc1 : ad 3b 1e 29 80 d0 03 4c  ec
0fc9 : 78 0e a9 01 8d 3c 1e d0  a8
0fd1 : ae ad 3b 1e 29 80 f0 ef  23
0fd9 : a9 02 8d 3c 1e d0 a0 a4  a2
0fe1 : d3 b1 d1 49 80 91 d1 60  c7
0fe9 : c9 20 b0 02 38 60 c9 e0  9e
0ff1 : b0 fa c9 bf 90 04 29 7f  55
0ff9 : 18 60 c9 a1 90 05 38 e9  ce
1001 : 40 18 60 c9 a0 f0 ef c9  83
1009 : 80 b0 e1 c9 40 b0 01 60  e1
1011 : c9 60 b0 03 29 bf 60 29  fb
1019 : df 18 60 aa ad 3b 1e 29  f1
1021 : 01 d0 04 8a 4c e9 0f e0  ef
1029 : 20 f0 f8 e0 30 90 bd e0  5c
1031 : 3a 90 f0 60 c9 20 b0 03  62
1039 : 09 40 60 c9 40 90 fb c9  c0
1041 : 60 b0 03 09 80 60 09 c0  8c
1049 : 60 8d 3b 1e 20 a0 0c ad  95
1051 : 3b 1e 29 01 d0 13 a0 00  2e
1059 : 84 45 a4 45 b1 f7 20 19  df
1061 : 0d e6 45 ce 39 1e d0 f2  ba
1069 : 60 a0 00 b1 f7 84 62 85  88
1071 : 63 a2 90 38 20 49 bc 20  d0
1079 : df bd a0 00 a2 03 b9 00  88
1081 : 01 f0 04 c8 ca d0 f7 30  88
1089 : 0d e0 00 f0 09 a9 20 20  c3
1091 : 19 0d ca 4c 8a 10 a2 00  21
1099 : bd 00 01 f0 06 20 19 0d  95
10a1 : e8 d0 f5 60 ad 3a 1e f0  82
10a9 : 18 a9 50 a2 1e a0 0f 20  62
10b1 : e2 0a d0 03 4c ae 08 ad  e3
10b9 : 30 1e c9 07 90 15 4c 19  61
10c1 : 09 ad 36 1e d0 0d a2 dc  ac
10c9 : a0 19 20 e7 0b 20 37 0c  a2
10d1 : 4c ae 08 20 1c 0c a2 9e  64
10d9 : a0 14 20 e7 0b 20 15 0c  a6
10e1 : 20 f7 13 20 e4 ff c9 03  41
10e9 : d0 03 4c 75 08 c9 86 d0  87
10f1 : 0b ad 3d 1e 49 01 8d 3d  33
10f9 : 1e 4c e1 10 c9 87 d0 0b  ea
1101 : ad 3d 1e 49 02 8d 3d 1e  bb
1109 : 4c e1 10 c9 88 d0 0b ad  1a
1111 : 3d 1e 49 80 8d 3d 1e 4c  94
1119 : e1 10 c9 89 d0 29 ad 3d  2d
1121 : 1e 49 40 8d 3d 1e 29 40  90
1129 : f0 b6 a2 3e a0 1e 86 f7  ea
1131 : 84 f8 a9 03 8d 39 1e a9  6b
1139 : 01 a2 1f a0 0f 20 5e 0e  ef
1141 : ad 3c 1e 30 d9 10 99 c9  b2
1149 : 8a d0 0b ad 3d 1e 49 10  be
1151 : 8d 3d 1e 4c e1 10 c9 85  5f
1159 : d0 89 ae 30 1e bd da 15  05
1161 : 10 02 a9 00 ae 31 1e 1d  04
1169 : 70 14 ae 32 1e 1d 78 14  aa
1171 : 8d 3c 03 ae 33 1e bd 73  b5
1179 : 14 09 00 8d 3d 03 ae 31  cd
1181 : 1e bd 7a 14 85 0d 20 ac  3a
1189 : 15 b0 0b a9 03 a2 02 a0  7d
1191 : 03 20 ba ff a9 02 a2 3c  01
1199 : a0 03 20 bd ff 20 c0 ff  7f
11a1 : ad 3a 1e d0 03 4c 2e 12  7c
11a9 : 20 84 15 ad 3f 1e f0 21  f1
11b1 : a9 01 a2 08 a0 02 20 ba  95
11b9 : ff a5 02 a2 50 a0 1e 20  22
11c1 : bd ff 20 c0 ff b0 23 a9  03
11c9 : 08 20 b4 ff a9 62 4c fb  e5
11d1 : 11 a6 02 e8 8a a2 50 a0  13
11d9 : 1e 20 bd ff a9 01 a2 08  b4
11e1 : a0 00 20 ba ff 20 c0 ff  e5
11e9 : 90 09 20 65 0d 20 37 0c  79
11f1 : 4c ae 08 a9 08 20 b4 ff  20
11f9 : a9 60 20 96 ff ad 3f 1e  54
1201 : d0 06 20 a5 ff 20 a5 ff  29
1209 : 20 b7 ff d0 dd ad 34 1e  77
1211 : c9 01 d0 35 a2 01 86 02  86
1219 : 20 a5 ff c6 02 d0 f9 a2  b8
1221 : 12 86 02 20 a5 ff c6 02  74
1229 : d0 f9 4c 4a 12 a2 80 a0  cc
1231 : 20 86 41 84 42 ad 34 1e  14
1239 : c9 01 d0 0d 18 a5 41 69  df
1241 : 12 85 41 a5 42 69 01 85  99
1249 : 42 ad 34 1e f0 07 c9 05  ab
1251 : d0 07 a9 8e 2c a9 5f 85  7a
1259 : 96 78 a9 36 85 01 20 b9  b1
1261 : 13 20 73 15 a2 03 20 c9  5a
1269 : ff a2 03 20 c6 ff 20 e8  3d
1271 : 15 ad 3d 1e 29 01 f0 05  d8
1279 : a9 02 20 a9 13 a9 ff 85  ea
1281 : 07 a9 00 85 08 20 e4 ff  23
1289 : c9 00 f0 10 48 20 38 15  21
1291 : 68 c9 12 d0 07 20 e4 ff  81
1299 : c9 11 d0 f9 ad 3a 1e d0  25
12a1 : 07 a0 00 b1 41 4c c7 12  e8
12a9 : a0 00 84 90 20 a5 ff a4  f5
12b1 : 90 f0 13 98 29 40 f0 03  f0
12b9 : 4c 5a 13 20 cc ff a9 03  75
12c1 : 20 c3 ff 4c eb 11 ac 34  af
12c9 : 1e f0 04 c0 05 d0 21 a4  1d
12d1 : 07 c8 c0 50 90 04 a0 00  22
12d9 : 84 08 84 07 c9 a0 f0 10  e9
12e1 : c9 20 d0 06 a4 08 f0 08  0e
12e9 : d0 47 c5 96 d0 02 e6 08  6a
12f1 : a8 ad 3d 1e 10 04 b9 30  eb
12f9 : 1d a8 2c 3d 1e 50 08 cc  3b
1301 : 3e 1e d0 03 4c 3e 13 ad  41
1309 : a1 02 29 01 d0 f9 98 25  9f
1311 : 0d 48 20 d2 ff 68 ae 30  03
1319 : 1e e0 0d f0 05 48 20 38  8c
1321 : 15 68 c9 0d d0 0b ad 3d  15
1329 : 1e 29 10 f0 04 a0 0a d0  0d
1331 : d6 a9 7f 8d 00 dc ad 01  0d
1339 : dc 29 80 f0 1c ad 3a 1e  3c
1341 : f0 03 4c 86 12 e6 41 d0  96
1349 : 02 e6 42 a5 41 cd 35 1e  97
1351 : d0 f0 a5 42 cd 36 1e d0  f4
1359 : e9 ad 3d 1e 29 01 f0 05  94
1361 : a9 03 20 a9 13 ad 3d 1e  99
1369 : 29 02 f0 05 a9 04 20 a9  ff
1371 : 13 a9 37 85 01 58 20 f6  19
```

Listing 1. »Convert 64« zur Datenübertragung zwischen C 64 und Amiga. Zur Eingabe ist das C 64-Programm »MSE« (siehe Text) nötig.

```
1379 : 15 20 ab ff a9 01 20 c3   34
1381 : ff 20 cc ff a9 03 20 c3   7e
1389 : ff a0 05 a2 00 20 a0 0c   0a
1391 : a2 28 a9 20 20 d5 0b a2   d8
1399 : 04 bd 7d 14 9d c8 04 ca   23
13a1 : 10 f7 20 37 0c 4c 75 08   a5
13a9 : 48 20 d2 ff 68 20 38 15   48
13b1 : ad a1 02 29 01 d0 f9 60   14
13b9 : 20 1c 0c a2 82 a0 14 20   fc
13c1 : e7 0b a2 00 bd 40 1e 20   6d
13c9 : 19 0d e8 e0 10 d0 f5 a2   64
13d1 : 8c a0 14 20 e7 0b a2 00   18
13d9 : a0 07 4c a0 0c 48 20 a0   e9
13e1 : 0c a2 04 86 02 68 d0 02   bb
13e9 : a2 00 bd 30 15 20 19 0d   d2
13f1 : e8 c6 02 d0 f5 60 a2 0f   e2
13f9 : a0 09 ad 3d 1e 29 01 20   a0
1401 : de 13 a2 0f a0 0b ad 3d   87
1409 : 1e 29 02 20 de 13 a2 1f   90
1411 : a0 0d ad 3d 1e 29 80 20   b8
1419 : de 13 a2 1f a0 0f 20 a0   52
1421 : 0c ad 3d 1e 29 40 d0 0c   07
1429 : a9 20 a2 06 20 19 0d ca   e0
1431 : d0 fa f0 2e a2 3e a0 1e   5b
1439 : 86 f7 84 f8 a9 03 8d 39   56
1441 : 1e a9 01 a2 1f a0 0f 20   3c
1449 : 4a 10 a9 20 20 19 0d 20   49
1451 : 19 0d ad 3e 1e 20 54 1b   8e
1459 : 90 02 a9 20 20 e9 0f 20   27
1461 : 19 0d a2 12 a0 11 ad 3d   af
1469 : 1e 29 10 20 de 13 60 40   ac
1471 : 20 00 00 20 60 a0 e0 00   24
1479 : 80 3f 7f ff 54 01 13 14   3a
1481 : 05 fd 00 05 44 01 14 05   cc
1489 : 09 20 ff 20 17 09 12 04   b0
1491 : 20 15 05 02 05 12 14 12   13
1499 : 01 07 05 0e ff fd 01 05   1f
14a1 : 44 01 14 05 09 20 13 05   f4
14a9 : 0e 04 05 0e fe fe 20 06   31
14b1 : 31 20 20 44 01 14 05 09   5a
14b9 : 20 13 05 0e 04 05 0e fe   05
14c1 : fe 20 06 33 20 20 53 54   b0
14c9 : 58 2f 45 54 58 fe fe 20   4e
14d1 : 06 35 20 20 45 4f 54 fe   9c
14d9 : fe 20 06 37 20 20 4b 0f   9e
14e1 : 0e 16 05 12 14 09 05 12   40
14e9 : 15 0e 07 13 14 01 02 05   85
14f1 : 0c 0c 05 fe fe 20 06 32   92
14f9 : 20 20 5a 05 09 03 08 05   33
1501 : 0e 20 15 0e 14 05 12 04   e0
1509 : 12 15 05 03 0b 05 0e fe   57
1511 : fe 20 06 34 20 20 4c 46   e8
1519 : 20 0e 01 03 08 20 43 52   14
1521 : fe fe 12 2f 13 20 20 5a   70
1529 : 15 12 15 05 03 0b ff 0e   d2
1531 : 05 09 0e 0a 01 20 20 c9   a5
1539 : 0d d0 03 4c a5 15 c9 0a   37
1541 : d0 03 4c 57 15 a4 d3 c0   d8
1549 : 28 d0 05 48 20 a5 15 68   78
1551 : 20 e9 0f 4c 19 0d a5 d6   f1
1559 : c9 18 d0 0e 20 14 1a a2   75
1561 : 27 a9 20 9d c0 07 ca 10   a8
1569 : fa 60 a6 d3 a4 d6 c8 4c   74
1571 : a0 0c a2 80 a0 1e 86 f7   d5
1579 : 84 f8 a2 80 a0 1f 86 f9   43
1581 : 84 fa 60 a9 50 a2 1e a0   a3
1589 : 0f 20 e2 0a ad 3f 1e f0   d1
1591 : 0c a2 03 c8 bd a1 15 99   39
1599 : 50 1e ca 10 f6 84 02 60   09
15a1 : 52 2c 53 2c a9 00 85 d3   bc
15a9 : 4c 57 15 ae 30 1e bd da   5d
15b1 : 15 30 02 18 60 29 7f aa   05
15b9 : bd d2 15 8d 3e 03 bd d3   71
15c1 : 15 8d 3f 03 a9 03 a2 02   0e
15c9 : a0 03 20 ba ff a9 04 38   18
15d1 : 60 3b 09 92 00 24 00 02   89
15d9 : 00 01 02 03 04 05 80 06   b1
15e1 : 07 08 09 82 0a 84 86 ad   b9
15e9 : 3a 1e d0 08 ad 0e dc 29   79
15f1 : fe 8d 0e dc 60 ad 0e dc   3a
15f9 : 09 01 8d 0e dc 60 a0 ba   71
1601 : 85 8d 81 ce fd 01 05 48   13
1609 : 01 15 10 14 0d 05 0e 15   77
1611 : 05 fe fe 20 06 31 20 20   04
1619 : 44 01 14 05 09 20 13 05   6c
1621 : 0e 04 05 0e fe fe 20 06   a9
1629 : 33 20 20 44 01 14 05 09   d4
1631 : 20 0c 01 04 05 0e fe fe   d3
1639 : 20 06 35 20 20 49 0e 08   42
1641 : 01 0c 14 13 16 05 12 1a   b6
1649 : 05 09 03 08 0e 09 13 fe   08
1651 : fe 20 06 37 20 20 50 01   0e
1659 : 12 01 0d 05 14 05 12 fe   80
1661 : fe 20 06 32 20 20 46 01   55
1669 : 12 02 05 0e fe fe 20 06   f4
1671 : 34 20 20 44 09 13 0b 02   9f
1679 : 05 06 05 08 0c 05 fe fe   a7
1681 : 20 06 36 20 20 50 12 0f   21
1689 : 07 12 01 0d 0d 05 0e 04   b5
```

```
1691 : 05 ff fd 00 16 49 08 12   05
1699 : 05 20 57 01 08 0c 3a 20   ae
16a1 : a0 ff fd 01 05 46 01 12   8b
16a9 : 02 05 0e fe fe 20 06 31   fd
16b1 : 20 20 52 01 08 0d 05 0e   af
16b9 : fe fe 20 06 33 20 20 48   45
16c1 : 09 0e 14 05 12 07 12 15   43
16c9 : 0e 04 fe fe 20 06 35 20   c0
16d1 : 20 53 03 08 12 09 06 14   06
16d9 : fe fe 12 2f 13 20 20 5a   28
16e1 : 15 12 15 05 03 0b ff 20   ae
16e9 : 20 35 30 20 20 20 20 37   a6
16f1 : 35 20 20 20 31 31 30 20   e0
16f9 : 20 20 31 33 34 2e 35 20   a6
1701 : 31 35 30 20 20 20 32 30   09
1709 : 30 20 20 20 33 30 30 20   0b
1711 : 20 20 36 30 30 20 20 31   bc
1719 : 32 30 30 20 20 31 38 30   40
1721 : 30 20 20 32 30 30 30 20   35
1729 : 20 32 34 30 30 20 20 33   60
1731 : 36 30 30 20 20 34 38 30   74
1739 : 30 20 20 fd 01 05 50 01   bd
1741 : 12 01 0d 05 14 05 12 fe   68
1749 : fe 20 06 31 20 20 52 53   f2
1751 : 20 32 33 32 20 45 09 0e   0a
1759 : 13 14 05 0c 0c 15 0e 07   e9
1761 : fe fe 20 06 33 20 20 5a   11
1769 : 05 09 03 08 05 0e 14 01   c8
1771 : 02 05 0c 0c 05 fe fe 20   ff
1779 : 06 35 20 20 54 01 02 05   85
1781 : 0c 0c 05 0e 20 0c 01 04   05
1789 : 05 0e fe fe 20 06 37 20   84
1791 : 20 54 01 02 05 0c 0c 05   47
1799 : 0e 20 13 10 05 09 03 08   33
17a1 : 05 12 0e fe fe 12 2f 13   76
17a9 : 20 20 5a 15 12 15 05 03   f7
17b1 : 0b ff fd 01 05 52 53 20   cc
17b9 : 32 33 32 20 45 09 0e 13   10
17c1 : 14 05 0c 0c 15 0e 07 fe   b8
17c9 : fe 20 06 31 20 20 42 01   8d
17d1 : 15 04 12 01 14 05 fe 20   f0
17d9 : 20 06 33 20 20 44 01 14   1e
17e1 : 05 0e 02 09 14 13 fe fe   63
17e9 : 20 06 35 20 20 53 14 0f   69
17f1 : 10 02 09 14 13 fe 20 20   2d
17f9 : 06 37 20 20 50 01 12 09   0e
1801 : 14 01 05 14 fe fe 12 2f   e8
1809 : 13 20 20 5a 15 12 15 05   c0
1811 : 03 0b ff fd 01 05 5a 05   05
1819 : 09 03 08 05 0e 14 01 02   d0
1821 : 05 0c 0c 05 ff 0b 05 09   4f
1829 : 0e 05 20 20 20 20 20 20   8a
1831 : 20 20 20 20 ff 15 0e 07   5c
1839 : 05 12 01 04 05 20 20 20   1a
1841 : 20 20 20 20 ff 07 05 12   ee
1849 : 01 04 05 20 20 20 20 20   56
1851 : 20 20 20 20 ff 38 2e 20   48
1859 : 44 01 14 05 0e 02 09 14   01
1861 : 20 3d 20 31 ff 38 2e 20   09
1869 : 44 01 14 05 0e 02 09 14   11
1871 : 20 3d 20 30 ff 00 10 20   be
1879 : 30 40 fd 01 05 44 01 14   07
1881 : 05 09 20 0c 01 04 05 0e   f5
1889 : fe fe 20 06 31 20 20 44   ec
1891 : 01 14 05 09 20 0c 01 04   6d
1899 : 05 0e fe fe 20 06 33 20   84
18a1 : 20 44 01 14 05 09 01 12   67
18a9 : 14 20 05 09 0e 13 14 05   04
18b1 : 0c 0c 05 0e fe fe 20 06   3b
18b9 : 35 20 20 fe fe 20 06 37   5e
18c1 : 20 20 44 01 14 05 09 14   d8
18c9 : 19 10 fe fe 12 2f 13 20   b1
18d1 : 20 5a 15 12 15 05 03 0b   42
18d9 : ff 28 53 14 01 12 2d 54   42
18e1 : 05 18 14 05 12 29 ff 28   53
18e9 : 56 09 1a 01 17 12 09 14   b9
18f1 : 05 29 20 20 ff 28 41 13   03
18f9 : 03 09 09 29 20 20 20 20   ac
1901 : 20 20 ff 28 54 05 18 14   2c
1909 : 0f 0d 01 14 29 20 20 20   b6
1911 : ff 28 54 05 18 14 0f 0d   53
1919 : 01 14 2b 29 20 20 ff 28   68
1921 : 4d 01 13 14 05 12 14 05   71
1929 : 18 14 29 20 ff 00 0e 1c   0a
1931 : 2a 38 46 fd 00 16 4e 01   b5
1939 : 0d 05 20 04 05 12 20 44   3b
1941 : 01 14 05 09 3a ff 20 1c   0b
1949 : 0c a2 14 a0 18 20 e7 0b   f8
1951 : ad 37 1e 20 66 1b a0 07   f5
1959 : 8c 38 1e 20 b7 1b a9 03   8e
1961 : 8d 39 1e a9 81 a2 08 ac   ee
1969 : 38 1e 20 5e 0e ad 3c 1e   00
1971 : 10 03 4c b1 08 f0 04 c9   f8
1979 : 02 d0 06 20 9e 1b 4c 5c   16
1981 : 19 ad 3c 1e c9 01 d0 06   38
1989 : 20 87 1b 4c 5c 19 c9 07   81
1991 : 90 0d 38 e9 07 aa ad 37   de
1999 : 1e 9d 70 1e 4c 5c 19 c9   05
19a1 : 05 b0 0d 38 e9 03 aa bd   26
```

```
19a9 : 70 1e 20 66 1b 4c 57 19   a1
19b1 : c9 05 d0 0c 38 ad 37 1e   bd
19b9 : e9 20 20 66 1b 4c 57 19   2b
19c1 : 18 ad 37 1e 69 20 20 66   26
19c9 : 1b 4c 57 19 a0 0f b9 3e   e9
19d1 : 03 20 35 10 99 3e 03 88   dc
19d9 : 10 f4 60 fd 00 16 46 05   0f
19e1 : 08 0c 05 12 3a 20 4b 05   4f
19e9 : 09 0e 05 20 44 01 14 05   e5
19f1 : 09 20 16 0f 12 08 01 0e   f3
19f9 : 04 05 0e 21 ff fd 01 05   26
1a01 : 54 01 02 05 0c 0c 05 0e   48
1a09 : 20 13 10 05 09 03 08 05   2a
1a11 : 12 0e ff a2 40 a0 05 86   a9
1a19 : 3d 84 3e a2 18 a0 05 86   24
1a21 : 3f 84 40 a9 11 85 02 a0   6e
1a29 : 27 b1 3d 91 3f 88 10 f9   17
1a31 : 18 a5 3d 69 28 85 3d a5   87
1a39 : 3e 69 00 85 3e 18 a5 3f   96
1a41 : 69 28 85 3f a5 40 69 00   0a
1a49 : 85 40 c6 02 d0 d9 60 a2   83
1a51 : 97 a0 07 86 3d 84 3e a2   01
1a59 : bf a0 07 86 3f 84 40 a9   67
1a61 : 11 85 02 a0 27 b1 3d 91   e2
1a69 : 3f 88 10 f9 38 a5 3d e9   a9
1a71 : 28 85 3d a5 3e e9 00 85   9e
1a79 : 3e 38 a5 3f e9 28 85 3f   99
1a81 : a5 40 e9 00 85 40 c6 02   3a
1a89 : d0 d9 60 20 1c 0c a2 fe   0d
1a91 : a0 19 20 e7 0b a2 34 a0   9b
1a99 : 19 20 e7 0b 20 3f 0c 10   6a
1aa1 : 03 4c b1 08 a9 3e a2 03   55
1aa9 : a0 0f 20 e2 0a f0 ed 20   55
1ab1 : cd 19 a9 40 8d 3c 03 a9   97
1ab9 : 3a 8d 3d 03 a9 01 a2 08   a7
1ac1 : a0 01 20 ba ff a5 02 18   a7
1ac9 : 69 03 a2 3c a0 03 20 bd   02
1ad1 : ff a2 30 a0 1d 86 f7 84   30
1ad9 : f8 a2 34 a0 1e a9 f7 20   93
1ae1 : d8 ff a5 90 f0 06 20 65   bf
1ae9 : 0d 20 37 0c 4c b1 08 fd   c4
1af1 : 01 05 54 01 02 05 0c 0c   3b
1af9 : 05 0e 20 0c 01 04 05 0e   ef
1b01 : ff 20 1c 0c a2 f0 a0 1a   01
1b09 : 20 e7 0b a2 34 a0 19 20   21
1b11 : e7 0b 20 3f 0c 10 03 4c   54
1b19 : b1 08 a9 3e a2 03 a0 0f   e3
1b21 : 20 e2 0a f0 ed 20 cd 19   9c
1b29 : a9 01 a2 08 a0 00 20 ba   fc
1b31 : ff a6 02 e8 8a a2 3e a0   19
1b39 : 03 20 bd ff a9 00 a2 30   41
1b41 : a0 1d 20 d5 ff 90 09 20   1b
1b49 : 65 0d 20 37 0c 4c 02 1b   85
1b51 : 4c b1 08 c9 20 b0 02 38   b1
1b59 : 60 c9 80 b0 01 60 c9 a0   4f
1b61 : 90 f5 c9 e0 60 48 8d 37   67
1b69 : 1e a9 12 85 02 a2 00 a0   08
1b71 : 07 20 a0 0c 20 8a 0a 20   f1
1b79 : dd 0c ee 37 1e c6 02 d0   c1
1b81 : f3 68 8d 37 1e 60 20 8a   6d
1b89 : 0a ce 37 1e ad 38 1e c9   35
1b91 : 08 b0 06 20 50 1a 4c 8a   93
1b99 : 0a ce 38 1e 60 20 8a 0a   21
1ba1 : ee 37 1e ad 38 1e a2 11   89
1ba9 : c9 18 90 06 20 14 1a 4c   07
1bb1 : 8a 0a ee 38 1e 60 a0 1d   a5
1bb9 : 18 ad 37 1e 69 30 85 f7   58
1bc1 : 90 01 c8 84 f8 60 fd 01   21
1bc9 : 05 44 09 13 0b 02 05 06   76
1bd1 : 05 08 0c 05 ff 20 1c 0c   08
1bd9 : a2 c7 a0 1b 20 e7 0b a2   9d
1be1 : 00 a0 16 20 a0 0c a9 3e   49
1be9 : 20 19 0d a2 3e a0 03 86   2f
1bf1 : f7 84 f8 a9 1e 8d 39 1e   0d
1bf9 : a9 00 a2 01 a0 16 20 5e   63
1c01 : 0e ad 3c 1e 10 03 4c 75   ee
1c09 : 08 a0 1d 20 cf 19 a9 3e   96
1c11 : a2 03 a0 0f 20 e2 0a f0   62
1c19 : bc a9 0f a2 08 a0 0f 20   c4
1c21 : ba ff a6 02 e8 8a a2 3e   af
1c29 : a0 03 20 bd ff 20 c0 ff   0f
1c31 : a9 0f 20 c3 ff 20 65 0d   93
1c39 : 20 37 0c 4c 75 08 20 20   da
1c41 : 20 20 20 20 20 20 20 20   41
1c49 : 20 20 20 20 20 20 20 20   49
1c51 : 20 20 20 20 20 20 20 20   51
1c59 : 20 20 20 20 01 15 13 20   db
1c61 : 53 10 05 09 03 08 05 12   c8
1c69 : 20 13 05 0e 04 05 0e 16   e3
1c71 : 0f 0e 20 44 09 13 0b 05   77
1c79 : 14 14 05 20 13 05 0e 04   76
1c81 : 05 0e a2 13 86 02 ad 3a   3c
1c89 : 1e d0 02 a2 00 bd 5d 1c   80
1c91 : 20 19 0d e8 c6 02 d0 f5   4a
1c99 : 60 10 12 11 13 05 11 a2   8b
1ca1 : 0f a0 0d 20 a0 0c a2 03   43
1ca9 : 86 02 ad 3f 1e d0 02 a2   39
1cb1 : 00 bd 9a 1c 20 19 0d e8   8b
1cb9 : c6 02 d0 f5 60 a2 03 a4   e4
```

```
1cc1 : 02 c8 bd 26 1d 99 50 1e   78
1cc9 : ca 10 f6 84 02 a9 01 a2   a0
1cd1 : 08 a0 02 20 ba ff a5 02   f4
1cd9 : a2 50 a0 1e 20 bd ff 20   bf
1ce1 : c0 ff 90 03 4c 68 0a a2   9b
1ce9 : 01 20 c6 ff a2 80 a0 20   9d
1cf1 : 86 f7 84 f8 20 cf ff a8   85
1cf9 : a5 90 f0 07 29 40 d0 10   fb
1d01 : 4c 68 0a 98 a0 00 91 f7   57
1d09 : e6 f7 d0 e8 e6 f8 d0 e4   7f
1d11 : 20 cc ff a9 01 20 c3 ff   ec
1d19 : a5 f7 8d 35 1e a5 f8 8d   d2
1d21 : 36 1e 4c a5 10 52 2c 53   19
1d29 : 2c 00 00 00 00 00 00 00   56
1d31 : 00 00 00 00 00 00 00 00   32
1d39 : 00 00 00 00 00 00 00 00   3a
1d41 : 00 00 00 00 00 00 00 00   42
1d49 : 00 00 00 00 00 00 00 20   8a
1d51 : 21 22 23 24 25 26 27 28   41
1d59 : 29 2a 2b 2c 2d 2e 2f 30   49
1d61 : 31 32 33 34 35 36 37 38   51
1d69 : 39 3a 3b 3c 3d 3e 3f 40   59
1d71 : 61 62 63 64 65 66 67 68   61
1d79 : 69 6a 6b 6c 6d 6e 6f 70   69
1d81 : 71 72 73 74 75 76 77 78   71
1d89 : 79 7a 5b 5c 5d 5e 0d 60   5f
1d91 : 00 00 00 00 00 00 00 00   92
1d99 : 00 00 00 00 00 00 00 00   9a
1da1 : 00 00 00 00 00 00 00 00   a2
1da9 : 00 00 00 00 00 00 00 00   aa
1db1 : 00 00 00 00 84 94 81 8b   bb
1db9 : 8e 99 9a dd 00 00 00 00   76
1dc1 : 00 00 00 00 00 00 00 00   c2
1dc9 : 00 00 00 00 00 00 00 00   ca
1dd1 : 00 00 00 00 00 00 00 00   d2
1dd9 : 00 00 00 00 01 ba 00 b9   33
1de1 : 00 00 00 00 00 00 00 00   e2
1de9 : 00 00 00 00 cf 00 00 ff   e6
1df1 : 41 42 43 44 45 46 47 48   e1
1df9 : 49 4a 4b 4c 4d 4e 4f 50   e9
1e01 : 51 52 53 54 55 56 57 58   f1
1e09 : 59 5a 00 00 7c 00 00 00   57
1e11 : 00 00 00 00 00 00 00 00   12
1e19 : 00 00 00 00 00 00 00 00   1a
1e21 : 00 00 00 00 00 00 00 00   22
1e29 : 00 00 00 00 00 00 00 06   36
1e31 : 02 01 02 00 00 00 00 00   35
1e39 : 00 80 00 00 00 00 a0 a0   3d
1e41 : a0 a0 a0 a0 a0 a0 a0 a0   40
1e49 : a0 a0 a0 a0 a0 a0 a0 00   07
1e51 : 00 00 00 00 00 00 00 00   52
1e59 : 00 00 00 00 00 00 00 00   5a
1e61 : 00 00 00 00 b4 b6 a0 94   0e
1e69 : 92 85 96 8e 8f c3 00 50   ed
1e71 : 43 00 00 00 00 00 00 00   b5
1e79 : 00 00 00 00 00 00 00 00   7a
```

Listing 1. Convert 64 Hauptprogramm

```
Name : standard               1d30 1e34
-----------------------------------------
1d30 : 40 61 62 63 64 65 66 67   00
1d38 : 68 69 6a 6b 6c 6d 6e 6f   28
1d40 : 70 71 72 73 74 75 76 77   30
1d48 : 78 79 7a 7b 7c 7d 7e 7f   37
1d50 : 20 21 22 23 24 25 26 27   40
1d58 : 28 29 2a 2b 2c 2d 2e 2f   48
1d60 : 30 31 32 33 34 35 36 37   50
1d68 : 38 39 3a 3b 3c 3d 3e 3f   58
1d70 : c0 41 42 43 44 45 46 47   e0
1d78 : 48 49 4a 4b 4c 4d 4e 4f   68
1d80 : 50 51 52 53 54 55 56 57   70
1d88 : 58 59 5a 5b 5c 5d 5e 5f   78
1d90 : 60 61 62 63 64 84 66 67   79
1d98 : 68 69 6a 6b 6c 6d 6e 6f   88
1da0 : 70 71 72 73 74 75 94 77   08
1da8 : 81 8e 99 9a 9e 7d 7e 7f   f9
1db0 : 80 81 82 83 84 84 94 81   c4
1db8 : 9e 8e 99 9a 8c 8d 8e 8f   e5
1dc0 : 90 91 92 00 00 00 00 00   be
1dc8 : 00 00 00 00 00 00 00 00   c9
1dd0 : 00 00 00 00 00 00 00 00   d1
1dd8 : 00 00 00 00 00 00 00 00   d9
1de0 : 00 00 00 00 00 00 00 00   e1
1de8 : 00 00 00 00 00 00 00 00   e9
1df0 : 00 00 00 00 00 00 00 00   f1
1df8 : 00 00 00 00 00 00 00 00   f9
1e00 : 00 00 00 00 00 00 00 00   01
1e08 : 00 00 00 00 0d 00 00 00   d9
1e10 : 00 00 00 00 00 00 00 00   11
1e18 : 00 00 00 00 00 00 00 00   19
1e20 : 00 00 00 00 00 00 00 00   21
1e28 : 00 00 00 00 00 00 00 00   29
1e30 : 0d 02 00 00 00 00 00 00   3f
```

Listing 2. Die Standard-Konvertierungstabelle wird automatisch geladen

```
Name : vizawrite              1770 1874
-----------------------------------------
1770 : 40 61 62 63 64 65 66 67   40
1778 : 68 69 6a 6b 6c 6d 6e 6f   68
1780 : 70 71 72 73 74 75 76 77   70
1788 : 78 79 7a 5b 5c 5d 5e 1f   2f
1790 : 20 21 22 23 24 25 26 27   80
1798 : 28 29 2a 2b 2c 2d 2e 2f   88
17a0 : 30 31 32 33 34 35 36 37   90
17a8 : 38 39 3a 3b 3c 3d 3e 3f   98
17b0 : 60 41 42 43 44 45 46 47   c0
17b8 : 48 49 4a 4b 4c 4d 4e 4f   a8
17c0 : 50 51 52 53 54 55 56 57   b0
17c8 : 58 59 5a 5b 5c 5d 5e 5f   b8
17d0 : 60 61 62 63 64 84 66 67   b9
17d8 : 68 69 6a 6b 6c 6d 6e 6f   c8
17e0 : 70 71 72 73 74 75 94 77   48
17e8 : 81 8e 99 9a e1 6f 6f 6f   a0
17f0 : 6f 6f 6f 7b 84 85 86 87   00
17f8 : 88 89 8a 8b 8c 8d 8e 8f   e7
1800 : 90 91 92 00 00 00 00 00   fe
1808 : 00 00 00 00 00 00 00 00   09
1810 : 00 00 00 00 00 00 00 00   11
1818 : 00 00 00 00 00 00 00 00   19
1820 : 00 00 00 00 00 00 00 00   21
1828 : 00 00 00 00 00 00 00 00   29
1830 : 00 00 00 00 00 00 00 00   31
1838 : 00 00 00 00 00 00 00 00   39
1840 : 00 00 00 00 00 00 00 00   41
1848 : 00 00 00 00 0d 00 00 00   19
1850 : 00 00 00 00 00 00 00 00   51
1858 : 00 00 00 00 00 00 00 00   59
1860 : 00 00 00 00 00 00 00 00   61
1868 : 00 00 00 00 00 00 00 00   69
1870 : 0d 02 00 00 30 ff 00 10   a2
```

Listing 3. Die Konvertierungstabelle für Vizawrite 64-Umwandlung

```
Name : startexter             1d30 1e34
-----------------------------------------
1d30 : 00 00 00 00 00 00 00 00   31
1d38 : 00 00 00 00 00 00 00 00   39
1d40 : 00 00 00 00 00 00 00 00   41
1d48 : 00 00 00 00 00 00 00 00   49
1d50 : 20 21 22 23 24 25 26 27   40
1d58 : 28 29 2a 2b 2c 2d 2e 2f   48
1d60 : 30 31 32 33 34 35 36 37   50
1d68 : 38 39 3a 3b 3c 3d 3e 3f   58
1d70 : 60 61 62 63 64 65 66 67   60
1d78 : 68 69 6a 6b 6c 6d 6e 6f   68
1d80 : 70 71 72 73 74 75 76 77   70
1d88 : 78 79 7a 5c 5d 5e 5f 00   2d
1d90 : 00 00 00 00 00 00 00 00   91
1d98 : 00 00 00 00 00 00 00 00   99
1da0 : 00 00 00 00 00 00 00 00   a1
1da8 : 00 00 00 00 00 00 00 00   a9
1db0 : 00 00 00 00 00 84 94 81   2a
1db8 : e1 8e 99 9a 00 00 00 00   9a
1dc0 : 00 00 00 00 00 00 00 00   c1
1dc8 : 00 00 00 00 00 00 00 00   c9
1dd0 : 00 00 00 00 00 00 00 00   d1
1dd8 : 00 00 00 00 00 00 00 00   d9
1de0 : 00 00 00 00 00 00 00 00   e1
1de8 : 00 00 00 00 00 00 00 00   e9
1df0 : 00 41 42 43 44 45 46 47   a0
1df8 : 48 49 4a 4b 4c 4d 4e 4f   e8
1e00 : 50 51 52 53 54 55 56 57   f0
1e08 : 58 59 5a 00 2d 00 00 00   76
1e10 : 00 00 00 00 00 00 00 00   11
1e18 : 00 00 00 00 00 00 00 00   19
1e20 : 00 00 00 00 00 00 00 00   21
1e28 : 00 00 00 00 00 00 00 00   29
1e30 : 0d 02 00 00 00 00 00 00   3f
```

Listing 4. Die Konvertierungstabelle für Star-Texter-Umwandlung

```
Name : textomat               1d30 1e34
-----------------------------------------
1d30 : 00 00 00 00 00 00 00 00   31
1d38 : 00 00 00 00 00 00 00 00   39
1d40 : 00 00 00 00 00 00 00 00   41
1d48 : 00 00 00 00 00 00 00 00   49
1d50 : 20 21 22 23 24 25 26 27   40
1d58 : 28 29 2a 2b 2c 2d 2e 2f   48
1d60 : 30 31 32 33 34 35 36 37   50
1d68 : 38 39 3a 3b 3c 3d 3e 3f   58
1d70 : 60 61 62 63 64 65 66 67   60
1d78 : 68 69 6a 6b 6c 6d 6e 6f   68
1d80 : 70 71 72 73 74 75 76 77   70
1d88 : 78 79 7a 5b 5c 5d 5e 0d   0b
1d90 : 60 61 62 63 64 65 66 67   80
1d98 : 68 69 6a 6b 6c 6d 6e 6f   88
1da0 : 70 71 72 73 74 75 76 77   90
1da8 : 78 79 7a 7b 7c 7d 00 00   9f
1db0 : 00 00 00 00 00 00 00 00   b1
1db8 : 00 00 00 00 00 00 00 00   b9
1dc0 : 00 00 00 00 00 00 00 00   c1
1dc8 : 00 00 00 00 00 00 00 00   c9
1dd0 : 20 94 00 00 00 84 81 e1   28
1dd8 : 99 8e 9a 00 00 00 00 00   5f
1de0 : 00 00 00 00 00 00 00 00   e1
1de8 : 00 00 00 00 00 00 00 3f   67
1df0 : 40 41 42 43 44 45 46 47   e0
1df8 : 48 49 4a 4b 4c 4d 4e 4f   e8
1e00 : 50 51 52 53 54 55 56 57   f0
1e08 : 58 59 5a db 2d 00 00 00   f2
1e10 : 00 00 00 00 00 00 00 00   11
1e18 : 00 00 00 00 00 00 00 00   19
1e20 : 00 00 00 00 00 00 00 00   21
1e28 : 00 00 00 00 00 00 00 00   29
1e30 : 0d 02 00 00 00 00 00 00   3f
```

Listing 5. Die Konvertierungstabelle für Textomat-Umwandlung

```
Name : textomat+              1d30 1e34
-----------------------------------------
1d30 : 00 00 00 00 00 00 00 00   31
1d38 : 00 00 00 00 00 00 00 00   39
1d40 : 00 00 00 00 00 00 00 00   41
1d48 : 00 00 00 00 00 00 00 00   49
1d50 : 20 21 22 23 24 25 26 27   40
1d58 : 28 29 2a 2b 2c 2d 2e 2f   48
1d60 : 30 31 32 33 34 35 36 37   50
1d68 : 38 39 3a 3b 3c 3d 3e 3f   58
1d70 : 60 61 62 63 64 65 66 67   60
1d78 : 68 69 6a 6b 6c 6d 6e 6f   68
1d80 : 70 71 72 73 74 75 76 77   70
1d88 : 78 79 7a 5b 5c 5d 5e 0d   0b
1d90 : 60 61 62 63 64 65 66 67   80
1d98 : 68 69 6a 6b 6c 6d 6e 6f   88
1da0 : 70 71 72 73 74 75 76 77   90
1da8 : 78 79 7a 7b 7c 7d 00 00   9f
1db0 : 00 00 00 00 00 00 00 00   b1
1db8 : 00 00 00 00 00 00 00 00   b9
1dc0 : 00 00 00 00 00 00 00 00   c1
1dc8 : 00 00 00 00 00 00 00 00   c9
1dd0 : 20 94 00 00 00 84 81 e1   28
1dd8 : 99 8e 9a 00 00 00 00 00   5f
1de0 : 00 00 00 00 00 00 00 00   e1
1de8 : 00 00 00 00 00 00 00 3f   67
1df0 : 40 41 42 43 44 45 46 47   e0
1df8 : 48 49 4a 4b 4c 4d 4e 4f   e8
1e00 : 50 51 52 53 54 55 56 57   f0
1e08 : 58 59 5a db 2d 00 00 00   f2
1e10 : 00 00 00 00 00 00 00 00   11
1e18 : 00 00 00 00 00 00 00 00   19
1e20 : 00 00 00 00 00 00 00 00   21
1e28 : 00 00 00 00 00 00 00 00   29
1e30 : 0d 02 00 00 00 00 00 00   3f
```

Listing 6. Die Konvertierungstabelle für Textomat+-Umwandlung

```
Name : mastertext             1d30 1e34
-----------------------------------------
1d30 : 60 61 62 63 64 65 66 67   20
1d38 : 68 69 6a 6b 6c 6d 6e 6f   28
1d40 : 70 71 72 73 74 75 76 77   30
1d48 : 78 79 7a 84 94 81 e1 00   89
1d50 : 20 21 22 23 24 25 26 27   40
1d58 : 28 29 2a 2b 2c 2d 2e 2f   48
1d60 : 30 31 32 33 34 35 36 37   50
1d68 : 38 39 3a 3b 3c 3d 3e 3f   58
1d70 : 40 41 42 43 44 45 46 47   60
1d78 : 48 49 4a 4b 4c 4d 4e 4f   68
1d80 : 50 51 52 53 54 55 56 57   70
1d88 : 58 59 5a 8e 99 9a 00 00   64
1d90 : 00 00 00 00 00 00 00 00   91
1d98 : 00 00 00 00 00 00 00 00   99
1da0 : 00 00 00 00 00 00 00 00   a1
1da8 : 00 00 00 00 00 00 00 00   a9
1db0 : 00 00 00 00 00 00 00 00   b1
1db8 : 00 00 00 00 00 00 0d 00   ed
1dc0 : 00 00 00 00 00 00 00 00   c1
1dc8 : 00 00 00 00 00 00 00 00   c9
1dd0 : 00 00 00 00 00 00 00 00   d1
1dd8 : 00 00 00 00 00 00 00 00   d9
1de0 : 00 00 00 00 00 00 00 00   e1
1de8 : 00 00 00 00 00 00 00 00   e9
1df0 : 00 00 00 00 00 00 00 00   f1
1df8 : 00 00 00 00 00 00 00 00   f9
1e00 : 00 00 00 00 00 00 00 00   01
1e08 : 00 00 00 00 00 00 00 00   09
1e10 : 00 00 00 00 00 00 00 00   11
1e18 : 00 00 00 00 00 00 00 00   19
1e20 : 00 00 00 00 00 00 00 00   21
1e28 : 00 00 00 00 00 00 00 00   29
1e30 : 0d 02 00 00 00 00 00 00   3f
```

Listing 7. Die Konvertierungstabelle für Master Text-Umwandlung

Veränderung des CLI

Mit dem Programm »Preferences« kann man einige Voreinstellungen für die Workbench vornehmen, die auch für das CLI gelten. Noch mehr Parameter lassen sich mit »InitCli3« verändern.

Will man den Amiga programmieren, muß man sich an das CLI gewöhnen. Neben den vielen Vorteilen, die CLI-Fenster bieten, gibt es aber auch einige Nachteile. Mit InitCli3 (Listing 1) kann das Fenster nach eigenen Wünschen verändert werden.

Es erscheint wenig sinnvoll, im CLI, das ja eine Programmierumgebung darstellt, vier Farben zur Verfügung zu haben. Vor allem wenn man weiß, daß man dadurch mehr als 20-KByte-Speicher mehr benötigt als bei zweifarbiger Darstellung. Dieses Problem kann mit InitCli3 leicht beseitigt werden, wenn man das Programm folgendermaßen aufruft:
initcli -S

Durch diesen Befehl wird eine der zwei Bit-Ebenen ausgeschaltet, was zur Folge hat, daß nur noch zwei Farben dargestellt werden können. Will man später zur vierfarbigen Darstellung zurückkehren, ist dies kein Problem:
initcli -s

Wer den Monochrom-Modus benutzen will und keinen Schalter an seinem Monitor besitzt, kann dies auch softwaremäßig tun. Mit
initcli -c
wird dieser Modus eingeschaltet. Der Aufruf, um wieder die Farben aus Preferences zu benutzen, lautet:
initcli -C

Ein weiterer interessanter Befehl von InitCli dient dazu, das CLI-Fenster auf die maximale Größe zu bringen. Das heißt, daß nach dem Aufruf
initcli -m
das Fenster den gesamten Bildschirm belegt. Einen Befehl zur Wiederherstellung der alten Größe gibt es jedoch nicht, aber dies läßt sich dann mit der Maus erledigen.

Sehr ärgerlich ist es, wenn Text mit 80 Zeichen pro Zeile dargestellt werden soll, aber durch den Fensterrahmen nicht alle Zeichen in eine Zeile passen. Dieses Problem umgeht man mit dem Aufruf:
initcli -B

Nach diesem Befehl wird der Rahmen des CLI-Fensters ausgeschaltet. Dadurch stehen für Text 80 Spalten und 32 Zeilen zur Verfügung. Die Titelzeile des Fensters wird noch gezeichnet, was durch Drücken von <CTRL L> verhindert werden kann. In diesem Modus kann das Fenster weder verschoben noch in der Größe verändert werden. Die Vorder- und Hintergrundschalter funktionieren aber immer noch, obwohl sie unsichtbar sind. Um wieder ein Fenster mit Rahmen zu erhalten, ruft man InitCli3 einfach so auf:
initcli -b

Mit dem Programm »Preferences« kann man zwar einstellen, ob der Workbenchscreen im Interlace-Modus (320 x 512 Punkte) arbeiten soll, aber diese Umstellung wird erst nach dem nächsten Booten des Amiga vorgenommen. InitCli3 stellt sofort nach dem Aufruf
initcli -I
den Interlace-Modus ein und mit
initcli -i
wieder aus.

Eine weitere interessante, bis jetzt noch nicht genutzte, Fähigkeit des Amiga ist es, verschiedene Zeichensätze auch in einem CLI-Fenster zu benutzen. Der Aufruf muß hierfür so aussehen:
initcli -F<Font>/<Größe>

Für <Font> muß ein Zeichensatzname eingegeben werden, der im Directory »Fonts« existiert. Mit Größe gibt man die gewünschte Zeichensatzgröße an, die natürlich in einem Subdirectory mit dem Fontname stehen muß. Fantastische Effekte kann man hier erreichen, wenn ein großer Zeichensatz verwendet wird (siehe Bild 1). Um wieder den normalen System-Zeichensatz zu benutzen, lautet der Aufruf:
initcli -f

Der letzte Parameter dient dazu, den ursprünglichen Zustand wiederherzustellen:
initcli -d

Danach sieht das Fenster aus wie vor allen Veränderungen, nur wenn das Fenster auf maximale Größe gebracht wurde, bleibt es auf dieser Größe.

Sollte man einmal die Parameter vergessen haben, erhält man mit
initcli ?
ein kurze Anleitung mit allen Parametern.

Um nun das Fenster ohne Rahmen und mit maximaler Größe darzustellen, muß man aber nicht InitCli3 zweimal aufrufen. Es können nämlich mehrere Parameter, durch Leerzeichen getrennt, angegeben werden:
initcli -B -S
Die Reihenfolge der Parameter spielt dabei keinerlei Rolle.

InitCli3 ist durch all seine Fähigkeiten ein Programm, daß sicher immer wieder seine Anwendung finden wird.(Torsten Jürgeleit/rb)

Programmname:	InitCli3
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	C
Compiler:	Aztec-C Version 3.40A
Aufrufe:	cc InitCli3 -s ln InitCli3.o -lc

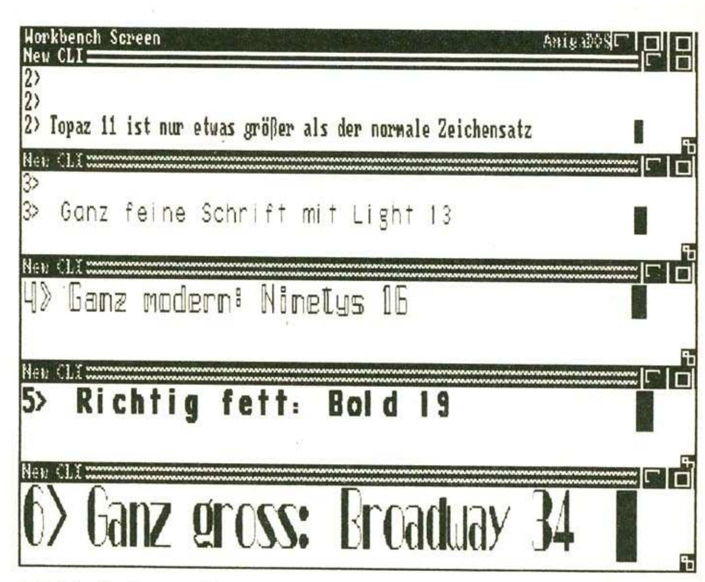


Bild 1. Mehrere CLI-Fenster mit anderen Zeichensätzen

```
Programm : InitCli3
-------------------

 1 /*****************************************************************/
 2 /*      InitCli3  Version 1.3  05/87        von Torsten Juergeleit  */
 3 /*                                                                 */
 4 /* Inklusive einer verbesserten Version von SOLOBITMAP            */
 5 /*                                                                 */
 6 /*      (Version fuer Aztec C Compiler 3.4a und KickStart 1.2)    */
 7 /*****************************************************************/
 8
 9
10 #include <exec/exec.h>
11 #include <devices/console.h>
```

Listing 1. Der Quellcode von InitCli3 für den Aztec-C-Compiler V3.40A und Kickstart 1.2

```
 12 #include <graphics/gfxbase.h>
 13 #include <graphics/display.h>
 14 #include <graphics/text.h>
 15 #include <intuition/intuition.h>
 16 #include <libraries/dos.h>
 17 #include <libraries/dosextens.h>
 18 #include <libraries/diskfont.h>
 19 #include <functions.h>
 20 #include <stdio.h>
 21
 22
 23 #define VERSION          "1.3"
 24 #define REV              0L
 25 #define NAMESIZE         40
 26 #define PSIZE            200L
 27 #define EOS              '\0'
 28 #define SYSTEMGADGETS    ( WINDOWDRAG | WINDOWSIZING )
 29 #define BUF_NAME         "Border"
 30
 31 /* wichtige Deklarationen und Definitionen von globalen Variablen */
 32
 33 struct GfxBase         *GfxBase       = NULL;
 34 struct ViewPort        *vp;
 35 struct ColorMap        *cm;
 36 struct BitMap          *b;
 37 struct IntuitionBase   *IntuitionBase = NULL;
 38 struct Preferences     *p;
 39 struct Window          *w             = NULL;
 40 struct DiskfontBase    *DiskfontBase  = NULL;
 41
 42 struct MsgPort iorp = {{ 0, 0, NT_MSGPORT, 0, 0 }, 0, -1, 0,
 43   { (struct Node *)&iorp.mp_MsgList.lh_Tail, 0,
 44     (struct Node *)&iorp.mp_MsgList.lh_Head, 0, 0 }};
 45 struct IOStdReq ior = {{{ 0, 0, 0, 0, 0 }, &iorp, 0 }, 0 };
 46
 47 UWORD mono_cm[] = { 0x000, 0x0f0, 0x000, 0x000 }; /* Farbwerte fuer Mon
    o- *
 48  * chrom-Darstellung :  *
 49  * schwarz, gruen,      *
 50  * schwarz, schwarz     */
 51
 52 VOID exit_program(), print_usage (), solobitmap_off(), solobitmap_on(),

 53  monochrom(), default_colors(), max_window(), border_on(), border_off()
   ,
 54  exit_border(), interlace_on(), interlace_off(), new_font(),
 55  default_font(), get_font();
 56
 57 struct Window *get_window();
 58
 59 SHORT check_flags();
 60
 61
 62  /* Haupt-Programm */
 63
 64 VOID
 65 main(argc, argv)
 66  SHORT argc;
 67  BYTE  *argv[];
 68 {
 69  REGISTER SHORT i;
 70
 71  if (!(GfxBase = (struct GfxBase *)
 72    OpenLibrary("graphics.library", REV))) exit_program(20);
 73  if (!(IntuitionBase = (struct IntuitionBase *)
 74    OpenLibrary("intuition.library", REV))) exit_program(21);
 75
 76    if (!(w = get_window())) exit_program(22);
 77    vp = &w->WScreen->ViewPort;
 78    b  = &w->WScreen->BitMap;
 79
 80    if (argc == 1) {
 81      printf("Flags fehlen !\n\n");
 82      exit_program(23);
 83    }
 84    if (*argv[1] == '?') print_usage();
 85    for (i=1; i<argc; i++) {
 86      if (*argv[i] != '-') {
 87        printf("Vermisse '-' vor dem Flag '%c' !\n\n", *(argv[i]+1));
 88      } else {
 89        switch(*(argv[i]+1)) {
 90          case 's' :
 91            solobitmap_off();
 92            break;
 93          case 'S' :
 94            solobitmap_on();
 95            break;
 96          case 'c' :
 97            default_colors();
 98            break;
 99          case 'C' :
100            monochrom();
101            break;
102          case 'm' :
103            max_window();
104            break;
105          case 'b' :
106            border_on();
107            break;
108          case 'B' :
109            max_window();
110            border_off();
111            break;
112          case 'i' :
113            interlace_off();
114            break;
115          case 'I' :
116            interlace_on();
117            break;
118          case 'f' :
119            default_font();
120            break;
121          case 'F' :
122            new_font(argv[i] + 2);
123            break;
124          case 'd' :
125            border_on();
126            solobitmap_off();
127            default_colors();
128            max_window();
129            interlace_off();
130            default_font();
131            break;
132          default :
133            printf("Unbekanntes Flag '-%c' !\n\n", *(argv[i]+1));
134            break;
135        }
136      }
137    }
138    exit_program(0);
139 }
140  /* Programm verlassen */
141
142 VOID
143 exit_program(error)
144  SHORT error;
145 {
146  if (ior.io_Device) {
147    if (iorp.mp_SigBit != -1) {
148      FreeSignal(iorp.mp_SigBit);
149    }
150    CloseDevice(&ior);
151  }
152  if (GfxBase)        CloseLibrary(GfxBase);
153  if (IntuitionBase)  CloseLibrary(IntuitionBase);
154  if (DiskfontBase)   CloseLibrary(DiskfontBase);
155  exit(error);
156 }
157  /* Bedienungshinweise */
158
159 VOID
160 print_usage()
161 {
162  printf("\n\x9b1;33mInitCLI\x9b0;3;31m Version %s  05/87", VERSION);
163  printf("\x9b0m  von \x9b4mTorsten Juergeleit\x9b0m\n\n");
164  printf(" \x9b7;42m Eingabeformat : \x9b0;40m\n");
165  printf("     InitCLI [-s] [-S] [-c] [-C] [-m] [-b] [-B] [-i] [-I] [-f]
    ");
166  printf(" [-Ffont/size]\n");
167  printf("              [-d]\n\n");
168  printf(" \x9b7;42m Bedeutung der Flags : \x9b0;40m\n");
169  printf("    -s              = SoloBitmap ausschalten\n");
170  printf("    -S              = SoloBitmap einschalten\n");
171  printf("    -c              = Farben aus Preferences einschalten\n");
172  printf("    -C              = Monochrom-Modus einschalten\n");
173  printf("    -m              = Maximale Window-Groesse\n");
174  printf("    -b              = Window-Rahmen einschalten\n");
175  printf("    -B              = Window-Rahmen ausschalten\n");
176  printf("    -i              = Interlace Modus ausschalten\n");
177  printf("    -I              = Interlace Modus einschalten\n");
178  printf("    -f              = Standard ROM-Zeichensatz einschalten\n")
    ;
179  printf("    -F[font]/[size] = Neuen Zeichensatz von Disk laden\n");
180  printf("    -d              = Standardeinstellungen einschalten\n\n");
181  exit_program(0);
182 }
183  /* Task mit dazugehoerigem Window suchen -          *
184    * entnommen aus 'SetFont 1.1' von Commodore-AMIGA */
185
186 struct Window *
187 get_window()
188 {
189  struct MsgPort          *con;
190  struct StandardPacket   *packet = NULL;
191  struct InfoData         *id     = NULL;
192  struct Window           *window = NULL;
193
194  if ((OpenDevice("console.device", -1L, &ior, 0L)) != NULL) exit_progra
```

```
     m(24);
195  if ((iorp.mp_SigBit = AllocSignal(-1L)) < 0) exit_program(25);
196
197  iorp.mp_SigTask = (struct Task *)FindTask(NULL);
198  if (iorp.mp_SigTask->tc_Node.ln_Type == NT_PROCESS) {
199    if (con = (struct MsgPort *)
200      ((struct Process *)iorp.mp_SigTask)->pr_ConsoleTask) {
201    if (packet = (struct StandardPacket *)
202      AllocMem((ULONG)sizeof(*packet), MEMF_CLEAR)) {
203    if (id = (struct InfoData *)
204      AllocMem((ULONG)sizeof(*id), MEMF_CLEAR)) {
205    packet->sp_Msg.mn_Node.ln_Name = (BYTE *)&(packet->sp_Pkt);
206    packet->sp_Pkt.dp_Link = (struct Message *)&(packet->sp_Pkt);
207    packet->sp_Pkt.dp_Port = &iorp;
208    packet->sp_Pkt.dp_Type = ACTION_DISK_INFO;
209    packet->sp_Pkt.dp_Arg1 = ((ULONG)id) >> 2; /* BPTR !@#$*&! */
210    PutMsg(con, packet);
211    WaitPort(&iorp);
212    window = (struct Window *)(id->id_VolumeNode);
213    FreeMem(id, (ULONG)sizeof(*id));
214  }
215  FreeMem(packet, (ULONG)sizeof(*packet));
216 }
217 }
218 }
219 ior.io_Unit = (struct Unit *) -1;
220 return(window);
221 }
222  /* Zweite Bitplane ausschalten */
223
224 VOID
225 solobitmap_on()
226 {
227  if (b->Depth == 2) {
228    b->Depth = 1;
229    FreeRaster(b->Planes[1], 640L, (ULONG)vp->DHeight);
230    RemakeDisplay();
231  }
232 }
233  /* Zweite Bitplane einschalten */
234
235 VOID
236 solobitmap_off()
237 {
238  UWORD    cm[4];
239  PLANEPTR planeptr;
240
241  if (b->Depth == 1) {
242    if (planeptr = AllocRaster(640L, (ULONG)vp->DHeight)) {
243      b->Depth     = 2;
244      b->Planes[1] = planeptr;
245      BltClear(planeptr, (ULONG)RASSIZE(640, vp->DHeight), 0L);
246      RemakeDisplay();
247      if (p = GetPrefs((struct Preferences *)
248        AllocMem(PSIZE, MEMF_CLEAR), PSIZE)) {
249      cm[0] = (UWORD)GetRGB4(vp->ColorMap, 0L);
250      cm[1] = (UWORD)GetRGB4(vp->ColorMap, 1L);
251      cm[2] = p->color2;
252      cm[3] = p->color3;
253      FreeMem(p, PSIZE);
254      FreeColorMap(vp->ColorMap);
255      vp->ColorMap = GetColorMap(4L);
256      LoadRGB4(vp, cm, 4L);
257    }
258  }
259 }
260 }
261  /* Monochrom-Modus einschalten */
262
263 VOID
264 monochrom()
265 {
266  FreeColorMap(vp->ColorMap);
267  vp->ColorMap = GetColorMap((ULONG)(2 * b->Depth));
268  LoadRGB4(vp, mono_cm, (ULONG)(2 * b->Depth));
269 }
270  /* Urspruengliche Farben aus Preferences einschalten */
271
272 VOID
273 default_colors()
274 {
275  UWORD cm[4];
276
277  if (p = GetPrefs((struct Preferences *)
278    AllocMem(PSIZE, MEMF_CLEAR), PSIZE)) {
279  cm[0] = p->color0;
280  cm[1] = p->color1;
281  cm[2] = p->color2;
282  cm[3] = p->color3;
283  FreeMem(p, PSIZE);
284  FreeColorMap(vp->ColorMap);
285  vp->ColorMap = GetColorMap((ULONG)(2 * b->Depth));
286  LoadRGB4(vp, cm, (ULONG)(2 * b->Depth));
287 }
288 }
289  /* Window auf Maximalgroesse bringen */
290
291 VOID
292 max_window()
293 {
294  ULONG deltax, deltay;
295
296  deltax = - (ULONG)w->LeftEdge;
297  deltay = - (ULONG)w->TopEdge;
298  MoveWindow(w, deltax, deltay);
299  deltax = (ULONG)w->WScreen->Width - (ULONG)w->Width;
300  deltay = (ULONG)w->WScreen->Height - (ULONG)w->Height;
301  SizeWindow(w, deltax, deltay);
302  Delay(10L);          /* warten, bis Window vergroessert wurde */
303 }
304  /* Window-Rahmen einschalten */
305
306 VOID
307 border_on()
308 {
309  BYTE *pointer_buf;
310  LONG *pointer;
311
312  pointer_buf = w->UserData;
313  pointer     = (LONG *)(pointer_buf + 6);
314  if (!(strcmp(pointer_buf, BUF_NAME))) {
315    fprintf(stdout, "\x0c\x1bc\n");  /* Bildschirm loeschen und */
316      /* Console-Device reinitialisieren */
317    w->UserData = NULL;
318    w->FirstGadget->NextGadget->NextGadget = (struct Gadget *)*pointer;
319    w->Title    = (UBYTE *)*++pointer;
320    w->Flags   |= SYSTEMGADGETS;      /* System-Gadgets einschalten */
321    w->Flags   &= ~BORDERLESS;        /* Rahmen einschalten */
322    RefreshWindowFrame(w);
323    FreeMem(pointer_buf, 14L);
324  }
325 }
326  /* Window-Rahmen ausschalten */
327
328 VOID
329 border_off()
330 {
331  BYTE  *pointer_buf;
332  LONG  *pointer;
333  BYTE  columns[4], lines[4];
334
335  if (w->UserData == NULL) {
336    max_window();
337    if (!(pointer_buf = (BYTE *)AllocMem(14L, MEMF_CLEAR)))
338      exit_program(26);
339    strcpy(pointer_buf, BUF_NAME);
340    pointer    = (LONG *)(pointer_buf + 6);
341    *pointer   = (LONG)w->FirstGadget->NextGadget->NextGadget;
342    *++pointer = (LONG)w->Title;
343
344    w->UserData = pointer_buf;
345    w->Title    = NULL;
346    w->FirstGadget->NextGadget->NextGadget = NULL;
347    w->Flags   &= ~SYSTEMGADGETS;      /* System-Gadgets ausschalten */
348    w->Flags   |= BORDERLESS;             /* Rahmen ausschalten */
349    RefreshWindowFrame(w);
350
351    fprintf(stdout, "\x9b0y\x9b0x\n");          /* Linke obere Ecke */
352    sprintf(&columns[0],"%d", (BYTE)((UWORD)640/w->IFont->tf_XSize));
353    fprintf(stdout, "\x9b%su\n", &columns[0]);
354    sprintf(&lines[0],"%d", (BYTE)((UWORD)w->WScreen->Height/
355      w->IFont->tf_YSize));
356    fprintf(stdout, "\x9b%st\x0c\n", &lines[0]);
357  }
358 }
359  /* Interlace Modus einschalten */
360
361 VOID
362 interlace_on()
363 {
364  GfxBase->system_bplcon0 |= INTERLACE;
365  RemakeDisplay();
366 }
367  /* Interlace Modus ausschalten */
368
369 VOID
370 interlace_off()
371 {
372  GfxBase->system_bplcon0 &= ~INTERLACE;
373  RemakeDisplay();
374 }
375  /* Neuen Zeichensatz von Disk laden */
376
377 VOID
378 new_font(name)
379  BYTE *name;
380 {
381  BYTE           font[NAMESIZE], *size;
382  REGISTER SHORT i;
383
384  if (*name == EOS) {
385    printf("Zeichensatzname und -groesse fehlen !\n\n");
386    exit_program(28);
387  }
```

```
388 size = name;
389 for (i=0; (*size != '/') && (*size != EOS); i++) font[i] = *size++;
390 if (!i) {
391   printf("Zeichensatzname fehlt !\n\n");
392   exit_program(29);
393 }
394 if ((*size == EOS) || (*++size == EOS)) {
395   printf("Zeichensatzgroesse fehlt !\n\n");
396   exit_program(30);
397 }
398 if (!(atol(size))) {
399   printf("Falsche Zeichensatzgroesse !\n\n");
400   exit_program(31);
401 }
402 font[i] = EOS;
403 strcat(font, ".font");
404 get_font(font, (UWORD)atoi(size), (UBYTE)FPF_DISKFONT);
405 }
406 /* Standard Zeichensatz aus dem ROM (topaz) einschalten */
407
408 VOID
409 default_font()
410 {
411 if (!(p = GetPrefs((struct Preferences *)
412   AllocMem(PSIZE, MEMF_CLEAR), PSIZE))) exit_program(32);
413 get_font("topaz.font", (UWORD)p->FontHeight, (UBYTE)FPF_ROMFONT);
414 FreeMem(p, PSIZE);
415 }
416 /* Zeichensatz umschalten */
417
418 VOID
419 get_font(font, size, flag)
420 BYTE  *font;
421 UWORD size;
422 UBYTE flag;
423 {
424 struct TextAttr     *ta, *old_ta;
425 struct TextFont     *tf, *old_tf;
426 BYTE                columns[4], lines[4];
427
428 if (!DiskfontBase) {
429   if (!(DiskfontBase = (struct DiskfontBase *)
430     OpenLibrary("diskfont.library", REV))) {
431     printf("'diskfont.library' fehlt !\n\n");
432     exit_program(33);
433   }
434 }
435 if (!(ta = (struct TextAttr *)
436  AllocMem((ULONG)sizeof(*ta), MEMF_CLEAR))) exit_program(34);
437 ta->ta_Name  = (STRPTR)font;
438 ta->ta_YSize = size;
439 ta->ta_Style = FS_NORMAL;
440 ta->ta_Flags = flag;
441 if (flag == FPF_DISKFONT) {
442  tf = (struct TextFont *)OpenDiskFont(ta);
443 } else {
444  tf = (struct TextFont *)OpenFont(ta);
445 }
446 if (!tf) {
447  printf("Kann Zeichensatz '%s/%d' nicht finden !\n\n", font, size);
448 } else {
449  old_ta          = w->WScreen->Font;
450  w->WScreen->Font = ta;
451  ta              = old_ta;
452  old_tf          = w->RPort->Font;
453  if (!(SetFont(w->RPort, tf))) old_tf = tf;
454  CloseFont(old_tf);
455  printf("\033c");       /* Bildschirm loeschen */
456 }
457 FreeMem(ta, (ULONG)sizeof(*ta));
458
459  if (!(strcmp(w->UserData, BUF_NAME))) {
460    fprintf(stdout, "\x9b0y\x9b0x\n");        /* Linke obere Ecke */
461    sprintf(&columns[0],"%d", (BYTE)((UWORD)640/tf->tf_XSize));
462    fprintf(stdout, "\x9b%su\n", &columns[0]);
463    sprintf(&lines[0],"%d", (BYTE)((UWORD)w->WScreen->Height/
464      tf->tf_YSize));
465    fprintf(stdout, "\x9b%st\x0c\n", &lines[0]);
466  }
467 }

(c) M&T
```

Listing 1. InitCli3 (Schluß)

Flexibles Wörterbuch

Zum Verwalten von großen Datenmengen bietet sich der Computer geradezu an. Eine sinnvolle Anwendung ist zum Beispiel das Lernen von Vokabeln.

Das Programm »Vokabel« auf dem Amiga nutzt dessen Fähigkeiten wie Pull-Down-Menüs und Gadgets (Schalter) und ist deshalb besonders benutzerfreundlich. Außerdem ist Vokabel in der Lage, mehrere Dateien zu verwalten und erweist sich somit als äußerst flexibel in der Anwendung. So können zum Beispiel für verschiedene Personen unterschiedliche Dateien angelegt und benutzt werden.

Da der Aufbau und die Bedienung von Vokabel sehr einfach sind, sollen die einzelnen Menüpunkte nur kurz erklärt werden.

Kurz nachdem das Programm gestartet wurde, erscheinen einige Angaben zum Status des Programms. Die »aktive Datei« ist diejenige, mit der gerade gearbeitet wird. Die Zahl der in dieser Datei vorhandenen Wörter wird bei »Anzahl der Vokabeln« angezeigt. »Fremdsprache« zeigt die Sprache an, für die die Wörter gedacht sind. »Datum« und »Uhrzeit« stimmen natürlich nur, wenn sie vorher mit »Preferences« eingestellt wurden, oder eine Echtzeituhr im Amiga eingebaut ist. Bei Sprechen wird dann noch angezeigt, ob die Sprachausgabe in Englisch aktiviert ist oder nicht.

Nach dem Laden sind einige der Menüpunkte in Schattenschrift (graue, unscharfe Schrift) dargestellt, da sie erst verwendet werden dürfen, wenn eine Datei geladen ist. Die folgenden Menüpunkte sind aber von Anfang an benutzbar.

Unter dem ersten Menütitel (»Programm«) finden sich nur zwei Punkte: »Neustart« und »Ende«. Mit Neustart wird das Programm einfach von Anfang an neu gestartet, bei Ende wird das Programm und auch Amiga-Basic verlassen. Natürlich fehlen bei beiden Funktionen nicht die Sicherheitsabfragen.

Im Menü »Sprechen« kann mit dem einzigen Menüpunkt »Englisch« eingestellt werden, ob die Worte auch vom Amiga gesprochen werden sollen. Leider funktioniert dies nur in Englisch. Wählt man diese Funktion an, müssen allerdings auf der Diskette von der gestartet wurde, folgende Dateien vorhanden sein:
system/say
devs/narrator.device
libs/translator.library

Diese Dateien benötigt der Amiga nämlich, um Sprache ausgeben zu können.

Vokabel spricht mehrere Sprachen

Wenn man sich das Pull-Down-Menü »Vokabeln« ansieht, steht dort als erstes der Menüpunkt »eingeben«, mit dem man in eine Eingaberoutine für die neuen Vokabeln gelangt. Hier wird als erstes abgefragt, um welche Fremdsprache es sich handelt. Möglich sind Englisch, Französisch, Lateinisch, Spanisch und Italienisch. Die meisten der Sonderzeichen für die verschiedenen Sprachen können wie gewohnt durch bestimmte Tastenkombinationen erreicht werden. Ein <e> mit accent aigu kann man erzeugen, indem man <'> drückt und danach ein <e>. In der Datei, in der dann die Vokabeln gespeichert werden, enthält auch die Sprache, so daß bei späteren Programmläufen die Sprache angezeigt werden kann. Hat man eine Sprache ausgesucht, kann man die Vokabeln eingeben, wobei für einen Fremdsprachenbegriff bis zu drei deutsche Wörter erlaubt sind. Bei dem Eingabefeld sollte man nicht über 40 Zeichen hinausgehen, da später nicht mehr dargestellt werden. Durch diese Länge ist man aber auch in der Lage, ganze Sprichwörter einzugeben und später abzufragen. Hat man dann genug Wörter eingetippt, wird die Funktion durch Drücken von <RETURN> bei der Eingabe beendet.

Durch die Eingabe der Vokabeln sind jetzt alle Menüpunkte anwählbar, auch der Punkt »ändern«. Hier lassen sich jetzt alle Vokabeln mit den Übersetzungen leicht nachträglich ändern. Auf dem Bildschirm sind nun außer der aktuellen Vokabel mit deutschen Begriffen auch fünf blaue Schalter zu sehen. Wie durch die Aufschrift schon klar wird, kann man weiter- und zurückblättern, einzelne Vokabeln löschen und ändern. Beim Ändern wird der Cursor durch einen Strich am unteren Rand dargestellt, will man den Begriff nicht editieren, drückt man einfach <RETURN>, und der Begriff wird unverändert übernommen. Ansonsten tippt man das neue Wort ein und schließt ebenfalls mit <RETURN> ab. Der letzte Schalter (»beenden«) funktioniert aber nur, wenn der Cursor nicht mehr sichtbar ist.

Mit »laden« kann man schon erstellte Dateien laden und Vokabeln üben. Sehr schön ist es dabei, daß die Sprache sowie die Anzahl der gespeicherten Worte angezeigt wird. Das Gegenstück zum Laden ist »speichern«, womit eingegebene Wörter in eine Datei mit frei wählbarem Namen geschrieben werden. Beide Funktionen können durch Drücken von <RETURN> bei der Eingabe des Dateinamens abgebrochen werden. Der letzte Menüpunkt unter »Vokabeln« ist »listen«, mit dem die Dateien im Dateiunterverzeichnis (Subdirectory) »Vokabeln« angezeigt werden.

Jetzt kommt der Teil des Programms, mit dem die eingegebenen Wörter wieder abgefragt werden. Als erstes sollen die Punkte unter dem Menütitel »Abfragen« erläutert werden. Hier hat man die Wahl zwischen Abfrage in der Fremdsprache, in Deutsch oder in beiden Sprachen. Beim »Wörterbuch« kann nun die gesamte Datei nach Wörtern durchsucht werden, wobei auch hier die Möglichkeit besteht, die Wörter in der Fremdsprache oder in Deutsch einzugeben.

Alles in allem ist Vokabel also ein Programm, mit dem das Pauken einer fremden Sprache einfach mehr Spaß macht.

(Rolf Kunisch/rb)

Programmname:	Vokabel
Computer:	A500, A1000, A2000 mit Kickstart 1.2, 512 KByte
Sprache:	Amiga-Basic
Bemerkung:	Für Sprachausgabe müssen die nötigen Programme auf der Boot-Diskette vorhanden sein (siehe Text)

```
Programm : vokabel
------------------

 1 ' *****************************
 2 ' **   Amiga-Vokabeltrainer   **
 3 ' **     AmigaBASIC  V1.2     **
 4 ' **      512 KByte RAM       **
 5 ' **      Rolf Kunisch        **
 6 ' *****************************
 7
 8 CLEAR ,40000&,4789
 9 RANDOMIZE TIMER
10 ON ERROR GOTO Errors ' Fehler abfangen
11 DIM b%(5986)
12 DEFINT i
13 maxindex=100 ' Dieser Wert kann je nach Speicherausbau verändert werden
14 Verboten1$=CHR$(27)  'ESC-Taste
15 Verboten2$=CHR$(127) 'DEL-Taste
16 Verboten3$=CHR$(139) 'HELP-Taste
17 Verboten4$=Verboten1$+Verboten2$+Verboten3$ 'alle drei Tasten
18 GOSUB Titel
19 GOSUB Hauptbildschirm
20 Uhr:
21 LOCATE 13,37:COLOR 5,0:PRINT TIME$
22 GOTO Uhr
23 ' *****************
24 ' * Vokabeleingabe *
25 ' *****************
26 VokEingabe:
27 IF Vokabelzaehler>0 THEN
28  index1=Vokabelzaehler+1
29 ELSE
```

Listing 1. Der Vokabeltrainer für den Amiga (Bitte ohne Zeilennummern eingeben)

```
30 GOSUB FrageSprache
31 index1=1
32 END IF
33 Fertig=0
34 WHILE Fertig=0 AND NOT index1>maxindex
35  CLS:PRINT
36  COLOR 3,0:PRINT TAB(31)"Vokabeln eingeben"
37  LINE (232,17)-(384,17),2
38  PRINT:PRINT
39  COLOR 4,0:PRINT " Bitte geben Sie die";:COLOR 2,0:PRINT STR$(index1);"
    . ";:COLOR 4,0:PRINT "Vokabel ein : "
40  COLOR 1,0:CALL Eingeben (Verboten4$,5,37+LEN(STR$(index1)))
41  IF Eingabe$="" THEN
42    Fertig=-1
43  ELSE
44    Vokabel$(index1)=Eingabe$
45    PRINT
46    FOR index2=1 TO 3
47      Eingabe:
48      COLOR 4,0:PRINT " Bitte geben Sie die";:COLOR 2,0:PRINT STR$(inde
        x2);". ";:COLOR 4,0:PRINT "Bedeutung ein : ";
49      COLOR 1,0:CALL Eingeben (Verboten4$,6+index2,40+LEN(STR$(index2)))

50      COLOR 4,0
51      IF index2=1 AND Eingabe$="" THEN LOCATE 7,1:GOTO Eingabe
52      IF Eingabe$="" THEN
53        index2=3
54      ELSE
55        Bedeutung$(index1,index2)=Eingabe$
56      END IF
57    NEXT index2
58    index1=index1+1
59  END IF
60 WEND
61 index1=index1-1
62 Vokabelzaehler=index1
63 RETURN
64 ' *****************
65 ' * Vokabelabfrage *
66 ' *****************
67 VokAbfrage:
68 GOSUB VokAbfrageFD
69 IF Vokzahl>0 THEN RETURN
70 FehlerGesamt=Fehlerzaehler
71 GOSUB VokabfrageDF
72 FehlerGesamt=FehlerGesamt+Fehlerzaehler
73 CLS:PRINT
74 COLOR 2,0:PRINT " Insgesamt";FehlerGesamt;"Fehler !"
75 CALL Pause (4)
76 RETURN
77
78 VokAbfrageFD:
79 Fehlerzaehler=0
80 Vokzahl=Vokabelzaehler
81 CALL ClearVokStatus (Vokabelzaehler)
82 WHILE Vokzahl>0
83  index1=0
84  WHILE Vokabelstatus(index1)=-1 OR (index1<1 OR index1>Vokabelzaehler)
85    index1=INT((RND*(Vokabelzaehler))+1)
86  WEND
87  CLS:COLOR 6,0:PRINT " Bitte übersetzen Sie :"
88  LINE (10,13)-(400,26),1,b
89  LINE (10,29)-(400,42),1,b
90  LOCATE 2,55:COLOR 4,0:PRINT "Vokabeln :";:COLOR 1,0:PRINT Vokzahl
91  LOCATE 3,55:COLOR 4,0:PRINT "Fehler   :";:COLOR 2,0:PRINT Fehlerzaehler
92  COLOR 3,0:LOCATE 3,3:PRINT Vokabel$(index1)
93  IF Speech$=Sprache$ THEN SAY TRANSLATE$(Vokabel$(index1))
94  UebersEingabe:
95  COLOR 4,0:xPos=5:yPos=3:CALL Eingeben(Verboten2$,5,3)
96  IF Eingabe$="" THEN UebersEingabe
97  IF Eingabe$=CHR$(27) THEN RETURN
98  COLOR 2,0:LOCATE 7,3
99  IF (Eingabe$<>Bedeutung$(index1,1)AND Eingabe$<>Bedeutung$(index1,2)AN
    D Eingabe$<>Bedeutung$(index1,3))OR Eingabe$=CHR$(139) THEN
100   IF Eingabe$<>CHR$(139) THEN
101     PRINT "Falsch !!"
102     PRINT
103   END IF
104   COLOR 4,0:PRINT "Die richtige Antwort ist : ";
105   COLOR 1,0:PRINT Bedeutung$(index1,1);
106   IF Bedeutung$(index1,2)="" THEN weiter1
107   IF Bedeutung$(index1,3)="" THEN
108     COLOR 4,0:PRINT  " oder ";
109     COLOR 1,0:PRINT  Bedeutung$(index1,2)
110   ELSE
111     COLOR 4,0:PRINT " , ";
112     COLOR 1,0:PRINT  Bedeutung$(index1,2);
113     COLOR 4,0:PRINT  " oder ";
114     COLOR 1,0:PRINT  Bedeutung$(index1,3)
115   END IF
116   weiter1:
117   CALL WarteTaste
118   Fehlerzaehler=Fehlerzaehler+1
119 ELSE
120   Vokabelstatus(index1)=-1
121   PRINT "Richtig !"
122   CALL Pause (1)
123   Vokzahl=Vokzahl-1
124  END IF
125 WEND
126 CALL Fehler (Fehlerzaehler)
127 IF Fehlerzaehler>2 THEN VokAbfrageFD
128 RETURN
129
130 VokabfrageDF:
131 Fehlerzaehler=0
132 Vokzahl=Vokabelzaehler
133 CALL ClearVokStatus (Vokabelzaehler)
134 WHILE Vokzahl>0
135  index1=0
136  WHILE Vokabelstatus(index1)=-1 OR (index1<1 OR index1>Vokabelzaehler)
137    index1=INT((RND*(Vokabelzaehler))+1)
138  WEND
139  CLS
140  COLOR 6,0:PRINT " Bitte übersetzen Sie :"
141  LINE (10,13)-(400,26),1,b
142  LINE (10,29)-(400,42),1,b
143  LOCATE 2,55:COLOR 4,0:PRINT "Vokabeln :";
144  COLOR 1,0:PRINT Vokzahl
145  LOCATE 3,55:COLOR 4,0:PRINT "Fehler   :";
146  COLOR 2,0:PRINT Fehlerzaehler
147  COLOR 3,0:LOCATE 3,3:PRINT Bedeutung$(index1,1)
148  Antworteingabe2:
149  COLOR 4,0:CALL Eingeben(Verboten2$,5,3)
150  IF Eingabe$="" THEN Antworteingabe2
151  IF Eingabe$=CHR$(27) THEN RETURN
152  LOCATE 7,3:COLOR 2,0
153  IF Eingabe$<>Vokabel$(index1) OR Eingabe$=CHR$(139) THEN
154    IF Eingabe$<>CHR$(139) THEN
155      PRINT "Falsch !!"
156      PRINT
157    END IF
158    COLOR 4,0:PRINT "Die richtige Antwort ist : ";
159    COLOR 1,0:PRINT Vokabel$(index1)
160    IF Speech$=Sprache$ THEN SAY TRANSLATE$(Vokabel$(index1))
161    CALL WarteTaste
162    Fehlerzaehler=Fehlerzaehler+1
163  ELSE
164    Vokabelstatus(index1)=-1
165    PRINT "Richtig !"
166    CALL Pause (1)
167    Vokzahl=Vokzahl-1
168  END IF
169 WEND
170 CALL Fehler (Fehlerzaehler)
171 IF Fehlerzaehler>2 THEN VokabfrageDF
172 RETURN
173 ' ******************
174 ' * Vokabeländerung *
175 ' ******************
176 VokAenderung:
177 index1=1
178 IF Vokabelzaehler=0 THEN RETURN
179 CLS:PRINT
180 COLOR 3,0:PRINT TAB(32)"Vokabeln ändern"
181 LINE (240,17)-(376,17),2
182 COLOR 4,5
183 LOCATE 5, 3:PRINT " weiterblättern "
184 LOCATE 5,23:PRINT " zurckblättern "
185 LOCATE 5,43:PRINT " löschen "
186 LOCATE 5,56:PRINT " ändern "
187 LOCATE 5,68:PRINT " beenden "
188 PRINT:PRINT
189 WHILE Vokabelzaehler>0
190 LOCATE 8,1:COLOR 2,0:PRINT STR$(index1);". ";:COLOR 4,0:PRINT "Vokabel
    : ";:COLOR 1,0:PRINT Vokabel$(index1);"                    ":PRINT
191 FOR index2=1 TO 3
192  COLOR 2,0:PRINT "  ";STR$(index2);". ";:COLOR 4,0:PRINT "Bedeutung : "
    ;:COLOR 1,0:PRINT Bedeutung$(index1,index2);"                    "
193 NEXT index2
194 WHILE MOUSE(0)>-1 OR (INT(MOUSE(2)/8)+1)<>5:WEND   'PRINT INT(MOUSE(2)/8
    ):WEND
195  x=INT(MOUSE(1)/8)+1
196  IF x=>3 AND x<=18 THEN
197    LOCATE 5,3:COLOR 5,4:PRINT " weiterblättern "
198    LOCATE 5,3:COLOR 4,5:PRINT " weiterblättern "
199    index1=index1+1
200    IF index1>Vokabelzaehler THEN index1=1
201  ELSEIF x=>23 AND x<=38 THEN
202    LOCATE 5,23:COLOR 5,4:PRINT " zurckblättern "
203    LOCATE 5,23:COLOR 4,5:PRINT " zurckblättern "
204    index1=index1-1
205    IF index1=0 THEN index1=Vokabelzaehler
206  ELSEIF x=>43 AND x<=51 THEN
207    LOCATE 5,43:COLOR 5,4:PRINT " löschen "
208    LOCATE 5,43:COLOR 4,5:PRINT " löschen "
209    CALL JaNeinRequester
210    IF Antwort$<>"n" THEN
211      FOR i=index1+1 TO Vokabelzaehler
212        Vokabel$(i-1)=Vokabel$(i)
213        FOR ii=1 TO 3
214          Bedeutung$(i-1,ii)=Bedeutung$(i,ii)
```

```
215         NEXT ii
216       NEXT i
217       Vokabelzaehler=Vokabelzaehler-1
218       IF index1>Vokabelzaehler THEN index1=Vokabelzaehler
219     END IF
220   ELSEIF x=>56 AND x<=63 THEN
221     LOCATE 5,56:COLOR 5,4:PRINT " ändern "
222     LOCATE 5,56:COLOR 4,5:PRINT " ändern "
223     xPos%=8:yPos%=13+LEN(STR$(index1))
224     COLOR 1,0:CALL Eingeben (Verboten4$,xPos%,yPos%)
225     IF Eingabe$<>"" THEN Vokabel$(index1)=Eingabe$
226     LOCATE 8,13+LEN(STR$(index1)):PRINT Vokabel$(index1)
227     FOR index2=1 TO 3
228       xPos%=9+index2:yPos%=19
229       CALL Eingeben(Verboten4$,xPos%,yPos%)
230       IF Eingabe$="" AND Bedeutung$(index1,index2)="" THEN
231         index2=3
232       ELSEIF Eingabe$<>"" THEN
233         Bedeutung$(index1,index2)=Eingabe$
234       END IF
235         LOCATE xPos%,yPos%:PRINT Bedeutung$(index1,index2)
236       NEXT index2
237     ELSEIF x=>68 AND x<=76 THEN
238       LOCATE 5,68:COLOR 5,4:PRINT " beenden "
239       LOCATE 5,68:COLOR 4,5:PRINT " beenden "
240       COLOR 1,0
241       RETURN
242     END IF
243 WEND
244 RETURN
245 ' *********************
246 ' * Vokabeln speichern *
247 ' *********************
248 VokSpeichern:
249 IF Vokabelzaehler=0 THEN RETURN
250 CLS:PRINT
251 COLOR 3,0:PRINT TAB(30)"Vokabeln speichern"
252 LINE (224,17)-(386,17),2
253 COLOR 4,0:PRINT:PRINT
254 PRINT " Bitte geben Sie den ";:COLOR 2,0:PRINT "Namen der Datei";:COLOR
     4,0:PRINT " ein , in die Sie die Vokabeln"
255 PRINT:PRINT " speichern wollen : ";:COLOR 1,0:CALL Eingeben (Verboten4$
    ,7,21)
256 IF Eingabe$="" THEN RETURN
257 Dateiname$="Vokabeln/"+Eingabe$+".voc"
258 OPEN "R",#1,Dateiname$,168
259  FIELD #1,40 AS a$,2 AS l1$,40 AS b$,2 AS l2$,40 AS c$,2 AS l3$,40 AS d
     $,2 AS l4$
260  FOR i=1 TO Vokabelzaehler
261    LSET a$=Vokabel$(i)
262    LSET b$=Bedeutung$(i,1)
263    LSET c$=Bedeutung$(i,2)
264    LSET d$=Bedeutung$(i,3)
265    LSET l1$=MKI$(LEN(Vokabel$(i)))
266    LSET l2$=MKI$(LEN(Bedeutung$(i,1)))
267    LSET l3$=MKI$(LEN(Bedeutung$(i,2)))
268    LSET l4$=MKI$(LEN(Bedeutung$(i,3)))
269    PUT #1,i
270  NEXT i
271 CLOSE #1
272 Dateiname$=LEFT$(Dateiname$,LEN(Dateiname$)-4)+".ind"
273 OPEN Dateiname$ FOR OUTPUT AS #1
274  WRITE #1,Vokabelzaehler
275  WRITE #1,Sprache$
276 CLOSE #1
277 Dateiname$=MID$(Dateiname$,10,LEN(Dateiname$)-13)
278 RETURN
279 ' *****************
280 ' * Vokabeln laden *
281 ' *****************
282 VokLaden:
283 CLS:PRINT
284 COLOR 3,0:PRINT TAB(33)"Vokabeln laden"
285 LINE (248,17)-(376,17),2
286 COLOR 4,0:PRINT:PRINT
287 PRINT " Bitte geben Sie den ";:COLOR 2,0:PRINT "Namen der Datei";:COLOR
     4,0:PRINT " ein , von der Sie die Vokabeln"
288 PRINT:PRINT " laden wollen : ";:COLOR 1,0:CALL Eingeben (Verboten4$,7,1,7)
289 IF Eingabe$="" THEN RETURN
290 Dateiname$="Vokabeln/"+Eingabe$+".ind"
291 OPEN Dateiname$ FOR INPUT AS #1
292  INPUT #1,Vokabelzaehler
293  INPUT #1,Sprache$
```

Listing 1. Vokabeltrainer für den Amiga
(Fortsetzung auf Seite 71)

Das AMIGA GLOSSAR

In der Dokumentation des Amiga stößt man immer wieder auf Begriffe, deren Bedeutung in den gerade verfügbaren Handbüchern nicht beschrieben wird. Ein Verzeichnis mit einer kurzen Beschreibung einzelner Stichworte hilft Ihnen hier aus der Klemme.

Ein großer Teil der Systembeschreibung des Amiga liegt bisher nur in englischer Sprache vor. Das ist besonders bedauerlich bei den vier Bänden des »Rom Kernel Manuals« (RKM). Aber auch viele Handbücher und Bedienungsanleitungen sind noch nicht in unsere Sprache übersetzt.

Dieses Glossar enthält die gängigsten Begriffe aus der Amiga-Literatur mit einer knappen, aber umfassenden Erklärung. Es kann die Dokumentation zum Computer natürlich nicht ersetzen, sondern soll als ergänzendes Nachschlagewerk dienen. Die Ausführungen sind nach den englischen Stichworten alphabetisch geordnet und beziehen sich bei den Themen aus den Rom Kernel Manuals, sofern nicht ausdrücklich auf die neuere Version hingewiesen wird, auf die Version 1.1 des Betriebssystems. Eine Neufassung für die Version 1.2 soll bereits verfaßt worden sein, ist aber — zu diesem Zeitpunkt — in den Staaten noch nicht veröffentlicht worden.

(David Twigg-Flesner/pa)

AbsExecBase Die Speicherstelle 4, der Sprungvektor zur -> exec.library (also die Adresse der Library), ist eine der empfindlichen Speicheradressen des Amiga. Da Programme erst über sie den Standort der Systemroutinen erfahren, führt ein Überschreiben dieser Adresse zum Systemabsturz.

ACBM (Amiga Contiguous Bitmap) Die »aneinandergrenzenden« Speicherbereiche der verschiedenen -> Bitplanes eines -> Bitmaps beim Amiga: Anders als beim -> IFF-Verfahren werden im Speicher des Amiga und bei den Dateien von Amiga-Basic die einzelnen Bitplanes getrennt der Reihe nach verwaltet beziehungsweise gespeichert. Umwandlungsprogramme für die beiden Formate sind auf -> FishDisk #27 zu finden.

AddMem Ein Hilfsprogramm für die Einbindung von Speichererweiterungen unter Version 1.1; anders als -> auto.config-Erweiterungen überprüft das Programm nicht, ob der zugewiesene Speicherbereich bereits belegt ist.

AGNUS oder AGNES Es scheint immer noch Unklarheit über die richtige Bezeichnung zu herrschen, aber da der Designer des Chips, Jay Miner, die erste Version benutzt, soll diese auch hier gelten. Agnus koordiniert als Adreßgenerator-Chip den Speicherzugriff sowohl der CPU als auch der Sonderchips (-> Chip-RAM, -> Fast-RAM) und ist für die Steuerung der -> DMA-Kanäle verantwortlich. Der Chip enthält die beiden Hilfschips -> Blitter und -> Copper.

alert Systemwarnung oder -> guru meditation nr.

aliasing (auch »the jaggies«) Die stufenähnliche Erscheinung bei nicht gerade senkrecht oder waagerecht verlaufenden Linien am Bildschirm.

aliasing distortion -> low pass filter, -> quantization

ALink Ein -> Linkeditor, der zum Zusammenbinden einzelner Codemodule (-> object code) zu einem lauffähigen Programm (-> load file) verwendet wird; war Teil der frühen -> developers kits; ist mittlerweile von der verbesserten Version, BLink (FishDisk #34), abgelöst worden.

Amiga-DOS (Disk Operating System) Die Bezeichnung der Laufwerk-, Dateiverwaltungs- und Multitaskingroutinen (-> process) des Amiga; diese wurden von der englischen Firma -> Metacomco aus Bristol in Anlehnung an das System »Tripos« geschrieben und sind weitgehend Unix-kompatibel. Die meisten Routinen sind in der Programmiersprache -> BCPL geschrieben. Die Grundroutinen von Amiga-DOS sind über die entsprechenden Befehle des -> CLI dem Benutzer zugänglich (siehe operating system).

AmigaLibDisk -> Fish-Disks

Amplitude modulation (AM) Die (fortlaufende) Veränderung der Höhe (Amplitude) eines Signals (einer Wellenform); das heißt die Lautstärke des Signals wird geändert. Dieses Verfahren wird beim Amiga benutzt, um beispielsweise die Intonationskurve bei der Sprachsynthese zu erzeugen; es kann auch für die Programmierungen von -> Hüllkurven eingesetzt werden. Das Verfahren wird beim Amiga mit -> attached channels realisiert (siehe frequency modulation) und ist von Amiga-Basic aus nicht ohne weiteres möglich.

Animation Eine besondere Stärke der Systemroutinen des Amiga ist die Animation, die in zwei verschiedenen Formen vorhanden ist: Sprite-Animation und Playfield-Animation (BOBs).

Unter Animation versteht man die Bewegung eines sich selbst in unterschiedlichen Bewegungsphasen befindlichen Objektes am Bildschirm (Beispiel: gehende Figur).

Der Amiga verfügt hierzu über acht Hardware-Sprites und eine beliebige Anzahl BOBs; anstelle der Hardware-Sprites erscheinen aber in der Regel sogenannte VSprites (virtual sprites = Scheinsprites). Hierbei handelt es sich um verschiedene grafische Formen eines mehrfach verwendeten Hardware-Sprites. Diese Mehrfachverwendung wird durch die Zuweisung eines eigenen -> DMA-Kanals zu jedem Hardware-Sprite ermöglicht; einzige Voraussetzung für die Mehrfachverwendung ist, daß die verschiedenen Formen des Sprites mindestens eine Bildschirmzeile Abstand voneinander haben müssen.

Ein BOB (Blitter OBject) ist ein Teil des normalen »Bildes«, das durch den -> Blitter sehr schnell von einer Position in eine andere »kopiert« werden kann.

Die Playfield-Animation liegt auch in zwei Formen vor; man kann mit den von Zeichentrickfilmen bekannten Verfahren der »cel animation« (-> sequenced drawing: Eine Folge von Einzelbildern, die ursprünglich auf Papier, später auf Zelluloid, gezeichnet wurden) arbeiten oder mit -> motion control die Geschwindigkeit und Beschleunigung eines BOBs oder Sprites in vertikaler und horizontaler Richtung steuern.

Die Animationsroutinen sind von Amiga-Basic über die entsprechenden Befehle erreichbar (siehe graphics primitives).

AnimComp / AnimObj -> GELs

ANSI X3.64-1979 (American National Standards Institute) Eine normgebende Institution der Vereinigten Staaten, die überwiegend Hardware-Normen aufstellt. X3.64-1979 ist eine Normierung der Steuercodes für -> Terminals und bildet die Grundlage der vom Amiga benutzten Tastatur- und Steuercodes; ergänzt werden die in der -> ISO / DIS 6429.2-Version implementierten Codes durch -> DEC-ähnliche und Amiga-spezifische Steuersequenzen.

applications program (Anwendungsprogramm) Jede Art von Programm, mit dem der Benutzer Daten verarbeiten kann (siehe utility).

argument template (Befehlszeilenmuster) Eine Anzeige der bei einem (meistens -> CLI-) Befehl erforderlichen Parameter sowie der Befehlssyntax am Monitor; eine »Help-Funktion«, die bei fast allen von Metacomco programmierten CLI-Befehlen durch Eingabe des Befehls mit einem Fragezeichen abgerufen werden kann. Bei Ergänzungsbefehlen aus anderen Quellen kann diese Funktion fehlen.

artifacting -> hue

ASCII (American Standard Code for Information Interchange) Eine standardisierte Codierung für Zeichensätze und Steuerzeichen (A=65,B=66...), die 1963 eingeführt wurde; mit diesem 7-Bit-Code ohne Paritätsvorschriften sind 128 Zeichen darstellbar und auch standardmäßig definiert. Das achte Bit wird in der Regel benutzt, um systemspezifische Zeichen zu codieren, wie das auch beim Amiga der Fall ist.

asynchronous (nicht zeitgleich) Wörtlich verstanden müßte diese Bezeichnung bedeuten, daß der Computer Befehle, Routinen und so weiter »nacheinander« ausführt, das heißt eine laufende Routine muß zuerst beendet sein, bevor die nächste Routine beginnen darf (siehe synchronous).
Im -> RKM wird diese Bezeichnung anscheinend anders ausgelegt, wodurch die Bedeutung ins Gegenteil umgekehrt wird, wie bei der Erläuterung des Befehls SAY im Amiga-Basic-Handbuch.

ATOM (ALink Temporary Object Modifier) Eine Hilfsroutine für Assembler, die dazu dient, -> Objektcode-Module neu zu linken, damit der erzeugte Code auf -> Chip-RAM (Daten) und -> Fast-RAM (Code) verteilt werden kann, was bei Erweiterungen über 512 KByte hinaus vor allem bei speicherintensiven Grafikanwendungen erforderlich sein kann. Um bereits gelinkte Programme zu modifizieren, müssen diese zuerst mit einem Hilfsprogramm wieder in nichtgelinkte Objectcode-Module zurückverwandelt und dann mit ATOM für einen neuen Linkvorgang vorbereitet werden.

attached channels (gekoppelte Audiokanäle) Eine Besonderheit des -> audio. device, die vier Kanäle paarweise (0/1, 1/2, 2/3) zu koppeln, um den höherwertigen (Daten-)Kanal durch den niederwertigen Kanal entweder in Amplitude oder Frequenz zu modulieren. Der Modulationsmodus wird durch Eingriff in die Hardware-Register bestimmt; das ADKCON-Register (&hDEF09E) enthält in den untersten 8 Bit die entsprechende Steuerung. Die Bits 0 bis 3 schalten die Amplituden-, 4 bis 7 die Frequenzmodulation des jeweiligen Kanals ein. Die Modulationsdaten werden für beide Modi als 16-Bit-Werte über den gesetzten Kanal verarbeitet, obwohl bei der Amplitudenmodulation nur die untersten 7 Bit gültig sind, weil die maximale Lautstärke 64 beträgt. Diese Festlegung auf 16 Bit ist notwendig, weil beide Modi gleichzeitig aktiv sein dürfen; in diesem Fall werden die Daten abwechselnd für die Modulation herangezogen. Bei Amiga-Basic ist dieses Verfahren durch POKEs erreichbar; dabei müssen die oberen 8 Bit des ADKCON-Registers unverändert bleiben, weil diese für das Laufwerk verwendet werden.

audio.device Eine Systemroutine (-> device), die vier Audiokanäle verwaltet und steuert. Zur Tonerzeugung ist eine Wellenform erforderlich, die als Datenwerte zu je 8 Bit in einem Integerfeld abgelegt wird. Digitalisierte Daten können auch verwendet werden, solange die Formatbestimmungen für die Daten eingehalten werden. Das audio.device wird auch in der Endstufe der -> Sprachsynthese benutzt (siehe attached channels). Über Amiga-Basic ist das audio.device mit den Befehlen SOUND und WAVE erreichbar (siehe sound synthesis).

auto.config Automatic configuration / automatische Einbindung von Zusatzgeräten wie Festplattenlaufwerke und Speichererweiterungen in das System: Solche Zusätze werden mit der -> startup-sequence, gegebenenfalls mit einem in der -> Expansion-Schublade abgelegten Treiber beim -> Booten des Systems angemeldet und erkannt. Für solche Geräte hat Commodore ein System von Erkennungsbytes eingeführt, um zu verhindern, daß ein gegebener Speicherbereich von mehreren Zusätzen gleichzeitig beansprucht werden kann. Zum Einbinden stehen insgesamt 128 Abschnitte (-> expansion slot) mit jeweils 4 KByte am oberen Ende des Speicherbereichs zur Verfügung; benötigt ein Zusatz mehr als diese 4 KByte, wie das beispielsweise bei einer über einen -> DMA-Kanal betriebenen Festplatte der Fall sein dürfte, so muß der für die normale Speichererweiterung vorgesehene Bereich benutzt werden (siehe Zorro).

backdrop (Hintergrundfenster) (-> layers) Eine besondere Art -> Window, beziehungsweise eine »Schicht« des Bildschirms, die stets die hinterste aller anderen »Schichten« bildet und die Funktion eines Hintergrunds bei verschiedenen anderen Programmen übernimmt (Beispiel: Workbench-Fenster). Als einziges -> Gadget besitzt ein solches Fenster das Schließ-Gadget. Ein »backdrop« ist von Amiga-Basic aus nicht definierbar.

backup Sicherheitskopie einer Diskette, Festplatte oder einer Datei

backward referencing -> forward referencing

base (Basisadresse) Die Einsprungsadresse einer -> Library, entweder im RAM-Speicher (bei diskettenresidenten) oder im ROM (-> WCS) (bei systemresidenten) Libraries. (siehe Abs ExecBase). In Amiga-Basic wird diese Adresse mit dem LIBRARY-Befehl ermittelt, damit die Libraryfunktionen abgerufen werden können.

batch file (auch command file, script, sequence file — Befehlslistendatei) Eine Textdatei, die -> CLI-Befehle enthält und mit dem EXECUTE-Befehl wie ein normaler CLI-Befehl abgerufen und ausgeführt wird (Beispiel: -> startup-sequence). Wie diese Datei werden »batch files« am besten im s-Verzeichnis der Diskette abgelegt, weil EXECUTE sie dort zuerst, danach erst im aktuellen Verzeichnis sucht.

BCPL (Binary Coded Programming Language) Ein Vorgänger der Programmiersprache C und daher auch mit C weitestgehend kompatibel; anders als C kennt BCPL nur einen Datentyp, das binäre Wort und ist daher zur Programmierung von Computersystemen, wie des -> Amiga-DOS, gut geeignet. Viele Bereiche von AmigaDOS, einschließlich der CLI-Befehle, sind nicht in C, sondern BCPL geschrieben. Ein entsprechender Compiler von Metacomco ist bereits erschienen.

Bimmer -> Blitter

Bitmap Bildschirmspeicher, in dem jedes Bit einem Punkt auf dem Bildschirm zugeordnet ist. Beim Amiga heißen die Bitmaps aber -> Bitplanes, weil das Bild »mehrschichtig« aufgebaut wird. In C ist »BitMap« ein Zeiger auf eine Bitplane-Struktur.

Bitmap Display Das »Zeichenblatt« für den Bildschirm; kann aus bis zu acht -> Bitplanes bestehen. Davon werden aber derzeit maximal fünf (sechs im HAM-Modus) unterstützt.

Bitplane Eine -> Bitmap für eine »Schicht« des Amiga-Bildschirms; die Anzahl der verwendeten Bitplanes bestimmt die Anzahl der verfügbaren Farben am Bildschirm (siehe colour registers). Die Bitplanes sind sehr speicherintensiv; eine Bitplane in niedrigauflösendem Modus braucht 8000 Byte; dieser Bedarf wird bei Hochauflösung oder -> Interlace verdoppelt, bei beiden gleichzeitig vervierfacht — ein Bildschirm mit Hochauflösung, Interlace und 16 Farben braucht also insgesamt 128000 Byte und ist daher bei einem Amiga mit nur 256 KByte nicht möglich, denn es bleiben nur etwa 30000 Byte für das Programm und die Daten. Bei dem PAL-Amiga mit 256/512 Bildschirmzeilen ist der Speicherbedarf entsprechend höher, und zwar um etwas über 25 Prozent (10240 Byte pro Bitplane im niedrigauflösenden Modus).

Blitter (auch Bimmer) Ein Hilfschip von -> Agnus, als

BLock Image TransferER (Blitter) verantwortlich für Datenverschiebungen (Bildschirmbereiche wie Fenster, aber auch Daten und Programme) im -> Chip-RAM; als Bit IMage ManipulatoR (Bimmer) kann Blitter über drei der eigenen -> DMA-Kanäle bis zu 256 logische Operationen (siehe minterm) an den Daten vornehmen, bevor sie über den vierten Kanal in den Speicher zurückgeschrieben werden.
Außerdem zeichnet der Chip Linien mit einer Geschwindigkeit von 1 Million Punkte pro Sekunde.

Blitter Object (BOB -> Animation) Teil eines -> Playfields, der von -> Blitter »animiert« wird; unterliegt in den Ausmaßen keinen Begrenzungen, wird aber möglicherweise im 32-Farben-Modus, bei gleichzeitiger Verwendung von -> Sprites in der Farbenzahl eingeschränkt (siehe colour registers).

.bmap files Besondere Dateien, die von Amiga-Basic für den LIBRARY-Befehl benötigt werden; diese Dateien werden mit dem Hilfsprogramm ConvertFD auf der Extras-Diskette aus den -> .fd-files erzeugt, damit Amiga-Basic die übergebenen Parameter bei Library aufrufen auf ihre Richtigkeit überprüfen kann. Das Hilfsprogramm läuft, zumindest unter Version 1.1, nicht völlig richtig; enthält ein .fd-file Funktionsnamen oder Wörter, die mit Schlüsselwörtern (FOR,PRINT...) des Amiga-Basic identisch sind, werden diese als -> Token statt als ASCII-Kette verschlüsselt. Eine korrigierte Version des Programms befindet sich auf FishDisk #27. Zum Glück sind die meist benötigten .fd-files bereits in .bmap-Format auf der Extras 1.2-Diskette enthalten.

boot disk Die Diskette, mit der das System initialisiert wird, also diejenige, die nach Anforderung der -> Workbench-Diskette eingelegt wird.

Boot-ROM (Start-ROM) Beim Einschalten des Amiga 1000 übernimmt dieser ROM-Baustein die Überprüfung des verfügbaren Speichers oder anderer Bausteine und fordert die Eingabe der -> Kickstart-Diskette; beim A500 und A2000 sind die Kickstart-Routinen auch im ROM untergebracht und werden nach der Systemüberprüfung automatisch aktiviert.

boot(strap) (wörtlich: Schnürsenkel) Die Initialisierungsroutinen eines Computersystems; abgeleitet von der Wendung »to pull oneself up by one's bootstraps«, die im Deutschen am besten mit der Geschichte des Barons von Münchhausen verglichen werden kann, der sich am eigenen Zopf aus dem Sumpf zog.
Beim Computer bedeutet das, daß der Computer mit einem Ladeprogramm sich selbst für die eigentliche Datenverarbeitung »initialisieren« muß; dies geschieht beim Amiga mit der Kickstart-Diskette (nur A1000) für das Betriebssystem selbst und mit einer »Workbench-Diskette« (-> system disk) für die Workbench. Da A500/A2000 den Kickstart im ROM haben, läuft die Initialisierung gleich nach der Überprüfung des -> Boot-ROM ab.

boot sectors (Startsektoren einer Diskette) Jede »selbststartende« Amiga-Diskette (Beispiel: Workbench) enthält diese beiden Sektoren, die das Einlesen der Diskettendaten (des Disk-Icons) ohne vorher eingelegte Workbench-Diskette erlauben. Bei frisch formatierten Disketten werden diese Sektoren mit dem -> CLI-Befehl INSTALL auf die Diskette geschrieben.

bridge board (auch bridge card) Die Bezeichnung für Zusatzkarten zum Amiga 2000, mit denen der Computer IBM-kompatibel gemacht wird; so genannt, weil die Karte wie eine Brücke einen Amiga-Slot mit einem PC-Slot verbindet (siehe Janus, Zorro).

buffer (Pufferspeicher) Ein Speicherbereich, der zwischen einem Programm und einem -> device oder einer Datei geschaltet wird und als Übergangsspeicher für den Datenaustausch benutzt wird. Die übliche Größe der vom System zugewiesenen Puffer beträgt 512 Byte. Jedes angeschlossene Laufwerk hat einen besonders großen Puffer, weil das -> trackdisk.device das sogenannte »track i/o-Verfahren« benutzt, also eine ganze Spur der Diskette in den Puffer liest. Da eine Spur 11 Sektoren zu je 512 Byte hat, vielleicht noch zusätzlich eine Menge Byte Müll enthält, werden etwa 17 KByte für jeden Diskpuffer verbraucht. Die Vergrößerung des Diskpuffers ist mit dem neuen (Version 1.2) CLI-Befehl ADDBUFFERS möglich. Ein Puffer kann dynamisch oder statisch sein; ersterer wird während eines Programmlaufs zugewiesen und nach Beendigung des Programms dem -> heap wieder zugefügt; letzterer bleibt bis zum Ausschalten des Computers zugewiesen und kann auch bei Nicht-Benutzung nicht mehr von anderen Routinen belegt werden (siehe dynamic allocation).

bug (Bazillus, daher (umgangssprachlich) Defekt, Macke) Ein in der Regel schwer auszumachender Fehler in einem Computerprogramm (siehe debug, troubleshooting).

C Eine Programmiersprache, die aus -> BCPL über B hervorgegangen ist. Mit ihr — sie enthält Aspekte sowohl einer höheren Programmiersprache als auch der Assembler-Sprache — wurde das Betriebssystem UNIX entwickelt. Die Standardbeschreibung der Sprache ist in dem Buch »Die Programmiersprache C« von Kernighan und Ritchie zu finden.

C:Directory (commands: Systemverzeichnis) Das Verzeichnis des -> Amiga-DOS; enthält die Befehle und Befehlsroutinen des -> CLI, die zuerst im Verzeichnis -> SYS:, dann hier gesucht werden (siehe logical device).

cache (auch cache-memory: Ablagespeicher) Ein (zusätzlicher) Speicherbereich, in dem Daten zwecks schnelleren Zugriffs vorübergehend abgelegt (wörtlich: »versteckt«) werden (siehe buffer).

CAD (computer aided design) Das technische Zeichnen am Computerbildschirm anstelle des Reißbretts.

CAM (computer aided manufacturing) Wie CAD, aber mit ausgemalten, dreidimensional wirkenden Darstellungen.

CAI (computer aided instruction) Computerunterstützter Unterricht (nicht Informatik).

CALL (computer aided language learning) computerunterstützter (Fremd-) Sprachenunterricht.

computer science Informatikunterricht.

case sensitivity Unterscheidung von Groß- oder Kleinschreibung in Wörtern. Vor allem bei Systemroutinen, die in der Assembler-Sprache aufgerufen werden, ist es wichtig, die richtige Schreibweise zu beachten. Die Routinen unterscheiden zwischen Groß- und Kleinschreibung und so werden sie beim Aufruf eventuell nicht richtig identifiziert. Bei C dient die Schreibweise zur Unterscheidung von Ein-/Ausgabeoperationen auf verschiedenen Systemebenen.

checkmark (Kontrollzeichen, »Haken«) Das Zeichen zur Markierung einer aktiven Attributoption in einem -> Menü.

Chip-RAM Derzeit die untersten 512 KByte des Speichers bei allen Amiga-Varianten; wird als Bildschirmspeicher, Stapelspeicher und Programmspeicher benutzt und enthält auch sämtliche Pufferspeicher. Puffer für -> DMA-Kanäle müssen hier liegen, weil dieser Bereich zur Zeit als einziger im System für die Sonderchips (daher CHIP-Ram) und insbesonders für den -> Blitter zugänglich ist (siehe Fast-RAM).

chromaticity (die »Farbigkeit« eines bestimmten Farbtons) Die Qualität der Farbe als Ergebnis der verschiedenen Anteile der Farben Rot, Grün und Blau. Je niedriger der Wert einer Farbe, um so dunkler der Anteil an der Farbigkeit und somit die Farbe selbst. Für die voreingestellten Farben des Amiga-Basic (siehe colour registers) sehen die entsprechenden Anteile folgender-

maßen (Reihenfolge R-G-B) aus:

0	0.0,	0.3,	0.6	16	0.0,	0.0,	0.0
1	1.0,	1.0,	1.0	17	0.8,	0.1,	0.1
2	0.0,	0.0,	0.0	18	0.0,	0.0,	0.0
3	1.0,	0.5,	0.0	19	1.0,	0.8,	0.6
4	0.0,	0.0,	1.0	20	0.2,	0.2,	0.2
5	1.0,	0.0,	1.0	21	0.3,	0.3,	0.3
6	0.0,	1.0,	1.0	22	0.4,	0.4,	0.4
7	1.0,	1.0,	1.0	23	0.4,	0.5,	0.5
8	0.4,	0.1,	0.0	24	0.5,	0.5,	0.5
9	0.90	0.3,	0.0	25	0.5,	0.6,	0.5
10	0.6,	1.0,	0.1	26	0.6,	0.6,	0.6
11	0.9,	0.7,	0.0	27	0.7,	0.7,	0.7
12	0.3,	0.3,	1.0	28	0.8,	0.7,	0.8
13	0.6,	0.1,	1.0	29	0.8,	0.8,	0.8
14	0.0,	1.0,	0.5	30	0.9,	0.9,	0.9
15	0.8,	0.8,	0.8	31	1.0,	1.0,	1.0

chunk Ein Datenblock in einer -> IFF-Datei, in dem eine bestimmte Art von Daten enthalten ist; zur Zeit sind die vier Haupttypen FORM, LIST, PROP und CAT sowie ein Leerblock definiert. Als Untertypen sind je nach Verwendungszwecken definiert: ILBM (Interleaved Bitmap) für Grafikdateien, FTXT (formatted text), 8SVX (8-Bit Sound Samples) und SMUS (Simple Music Score). Diese sind wiederum in untergeordnete chunk-Typen aufgeteilt:

ILBM: BMHD (bitmap header), CMAP (colour map), GRAB (hotspot, der aktive Punkt des Mauszeigers oder eines »brush«), T (destination merge data), SPRT (Sprite data), CAMG (Amiga Viewport mode) und BODY (standard data chunk — enthält die eigentlichen Daten); hierzu kommen einige »privat» definierte chunks, wie CRNG (colour range DPaint) und CCRT (colour cycling range / timing Graphicraft)

FTXT:CHRS (characters nach -> ANSI/ISO), FONS (font specifiers), BODY

SMUS:SHDR (score header), NAME (des Stücks), (c), AUTH (Author), ANNO (Annotation zum Stück), I (instrument specifier), TRAK (Daten), BODY; IREV ist ein »privater« chunk bei Instant Music

8SVX:VHDR (header), NAME bis ANNO wie bei SMUS, ATAK und RLSE (Attack / Release der Hüllkurve) und BODY.

Diese chunk-Typen werden durch weitere, privat definierte Typen, ständig ergänzt.

cleanup (Speicheraufräumung) Die von einem -> process zu erledigenden »Aufräumungsarbeiten«, bevor alle zum process zugehörigen -> Tasks von den verschiedenen Systemlisten »abgekoppelt« und dadurch effektiv gelöscht werden; hierzu gehört beispielsweise die Beantwortung aller noch nicht verarbeiteten -> messages, die Freigabe von Puffern und statischen Speichern, der Aufruf der Systemroutinen für die Freigabe zugewiesenen Speicherplatzes, kurzum die Rückgabe aller von dem Programm beanspruchten -> resources.

CLI (Command Line Interface: Befehlszeilen-Eingabe von -> Amiga-DOS) Ein besonderes -> Window (Fenster), mit dem der Programmierer Zugang zu den Amiga-DOS-Routinen erhält; diesen Zugang kann man auf verschiedene Weise erreichen: a) mit -> Preferences das CLI einschalten, die SYSTEM-Schublabe öffnen und das CLI-Icon doppelt anklicken; b) während des Einlesens der Workbench-Diskette durch Unterbrechung der Ausführung der »startup-sequence« mit der Tastenkombination <CTRL D> nach Erscheinen des großen CLI-Windows; c) das Löschen oder Umbenennen der Datei »startup-sequence« (beispielsweise in »no-startup-sequence«; d) die »startup-sequence« dahingehend editieren, daß neben dem Workbench-Fenster ein CLI-Window gleich mitaktiviert wird. Jedes CLI-Window entspricht einer getrennten Task, effektiv einem weiteren -> Terminal, das im Vorder- oder Hintergrund laufen kann. Bis zu maximal 20 solche CLIs dürfen gleichzeitig in Betrieb sein, vorausgesetzt, genügend Speicherplatz ist dafür verfügbar. Ein CLI kann entweder »interaktiv« oder »nicht-interaktiv« sein; im ersten Fall werden eingegebene Befehle direkt ausgeführt, die Ausgabe erscheint im selben Fenster; im zweiten Fall läuft das CLI »im Hintergrund«, weil mit RUN oder RUN EXECUTE gestartet und hat nur ein Ausgabe-Fenster. Die Eingabe erfolgt dann entweder über die Parameterangabe nach RUN oder aus dem mit EXECUTE aufgerufenen -> batch file. Da jeweils nur ein CLI-Window »aktuell« sein kann, stellt Amiga-DOS auch Befehle für die Überwachung und den Abbruch der laufenden CLIs (STATUS, BREAK) zur Verfügung (siehe TCB). Programme, die vom CLI aus gestartet werden (einschließlich der CLI-Befehle selbst), laufen alle als solche »Tochterprozesse« ab; während der Ausführung »schläft« das aufrufende CLI, bis es durch Beenden des aufgerufenen Programms wieder »geweckt« wird.

clipboard.device Eine Systemroutine, die für die Verwaltung und Ablage von »clips« (Beispiel: mit der Kamera bei Textcraft kopierte Textstellen) zuständig ist. Solche »clips« können entweder im Arbeitsspeicher des Programms, im clipboard-Speicher oder in einer Datei im clipboard-Verzeichnis einer Diskette aufbewahrt werden. Clips müssen in IFF-Format (FTXT oder ILBM) abgelegt werden. Obwohl die bisherigen Programme in der Regel nur ein Clip auf einmal verwalten, können vom System durchaus mehrere benutzt werden.

clipping (abschneiden) Beschränkung des Bereiches der Bildschirmausgabe auf ein -> ViewPort. Clipping ist besonders bei der Verwendung von -> Bitmaps erforderlich, um zu verhindern, daß Daten (versehentlich) außerhalb des entsprechenden Bitmap-Speichers geschrieben werden, was in vielen Fällen zu einem Systemabsturz führen dürfte.

CList.library Systemlibrary für die Verwaltung von Zeicheneingaben.

clone (Zweig, Ableger) Die (gängige) Bezeichnung für einen Computer, der dem IBM-PC angelehnt ist; der -> Sidecar-Zusatz des Amiga entspricht einem solchen »Ableger«.

collision stack (Speicher des Amiga-Basic für die Ablage von Kollisionsmeldungen): Maximal 16 Einträge sind möglich, alle weiteren werden ignoriert.

colour descriptor words Zwei (16-Bit-)Datenworte, die eine Zeile eines Sprites definieren; die gesetzten Pixel bestimmen die Gestalt und die Farbe.

colour indirection (indirekte Farbbestimmung) Die Bildschirmdarstellung des Amiga besteht aus -> Bitplanes. Für jeden Punkt der jeweiligen Darstellung (Hintergrund, Fenster, Sprites, ...) ist in den Bitplanes ein Bit reserviert. Aus den Informationen »Bit gesetzt (1) oder nicht gesetzt (0)« und der binären Wertigkeit der Plane (1,2,4,8, 16) ermittelt der Amiga die Nummer eines Farbregisters, das dann die entsprechende Farbe enthält (Beispiel mit drei Bitplanes: 0*1 (Bit in Plane mit Wertigkeit 1 nicht gesetzt) +1*2+1*4 = Farbregister 6). Mit dieser Technik kann es bei der gleichzeitigen Verwendung von BOBs und Sprites mit mehr als 16 Farben zu einer Änderung der BOB-Farben kommen, weil die Farbregister 16 bis 31 von den Sprites genutzt werden.

colour map Der Teil einer -> ViewPort-Definition, der die Rot/Grün/Blau-Werte der Farbregister enthält. In Amiga-Basic wird diese Liste mit dem PALETTE-Befehl erzeugt.

colour register Der Amiga hat 32 Farbregister mit jeweils 16 Bit. Davon werden insgesamt 12 Bit für die Definition der Rot/Grün/Blau-Anteile der jeweiligen Farbe benutzt, wodurch sich insgesamt 4096 Farben definieren lassen. Beim Aufruf von Amiga-Basic werden diese Register mit den folgenden Farben voreingestellt:

0	dunkelblau	16	schwarz
1	weiß	17	rot
2	schwarz	18	schwarz
3	orange	19	hellbraun
4	blau	20	dunkelgrau
5	purpur	21	dunkelgrau
6	zyan	22	dunkelgrau
7	weiß	23	dunkelgrau
8	blau	24	mittelgrau
9	orange	25	mittelgrau
10	hellgrün	26	mittelgrau
11	goldgelb	27	mittelgrau
12	blau	28	hellgrau
13	lila	29	hellgrau
14	grünblau	30	hellgrau
15	mittelgrau	31	weiß

Die letzten 16 Register sind ebenfalls, jeweils in Vierergruppen, paarweise den acht Sprites zugewiesen. Als Sprite-Farbregister ist das erste Register der Gruppe stets mit der Hintergrundfarbe belegt und daher »transparent«. Die voreingestellten Farben der Register 20 bis 31 stellen eine Abstufung der Farben und nicht identische Grautöne dar. (siehe colour indirection, chromaticity).

Fortsetzung AMIGA 10/87

```
294 CLOSE #1
295 Dateiname$=LEFT$(Dateiname$,LEN(Dateiname$)-4)+".voc"
296 OPEN "R",#1,Dateiname$,168
297 FIELD #1,40 AS a$,2 AS 11$,40 AS b$,2 AS 12$,40 AS c$,2 AS 13$,40 AS d
    $,2 AS 14$
298 FOR i=1 TO Vokabelzaehler
299   GET #1,i
300   Vokabel$(i)=LEFT$(a$,CVI(11$))
301   Bedeutung$(i,1)=LEFT$(b$,CVI(12$))
302   Bedeutung$(i,2)=LEFT$(c$,CVI(13$))
303   Bedeutung$(i,3)=LEFT$(d$,CVI(14$))
304 NEXT i
305 CLOSE #1
306 Dateiname$=MID$(Dateiname$,10,LEN(Dateiname$)-13)
307 RETURN
308 ' *************
309 ' * Wörterbuch *
310 ' *************
311 WoerterbuchFD:
312 CLS:PRINT
313 COLOR 3,0:PRINT TAB(35)"Wörterbuch"
314 LINE (264,17)-(360,17),2
315 PRINT:PRINT
316 COLOR 4,0:PRINT "Bitte geben Sie den ";:COLOR 2,0:PRINT "Suchbegriff";:
    COLOR 4,0:PRINT " ein : ";:COLOR 1,0
317 CALL Eingeben(Verboten4$,5,39)
318 IF Eingabe$="" THEN RETURN
319 gefunden=0
320 FOR i=1 TO Vokabelzaehler
321  IF Eingabe$=Vokabel$(i)THEN
322    COLOR 2,0:PRINT:PRINT Vokabel$(i);:COLOR 1,0:PRINT " - ";:COLOR 3,0:
       PRINT Bedeutung$(i,1);
323    IF Bedeutung$(i,2)<>"" THEN COLOR 1,0:PRINT ",";:COLOR 3,0:PRINT Bed
       eutung$(i,2);
324    IF Bedeutung$(i,3)<>"" THEN COLOR 1,0:PRINT ",";:COLOR 3,0:PRINT Bed
       eutung$(i,3);
325    gefunden=1
326  END IF
327 NEXT i
328 IF gefunden=0 THEN COLOR 2,0:PRINT:PRINT " Suchbegriff nicht gefunden !
    !"
329 CALL WarteTaste
330 GOTO WoerterbuchFD
331
332 WoerterbuchDF:
333 CLS:PRINT
334 COLOR 3,0:PRINT TAB(35)"Wörterbuch"
335 LINE (264,17)-(360,17),2
336 PRINT:PRINT
337 COLOR 4,0:PRINT "Bitte geben Sie den ";:COLOR 2,0:PRINT "Suchbegriff";:
    COLOR 4,0:PRINT " ein : ";:COLOR 1,0
338 CALL Eingeben(Verboten4$,5,39)
339 IF Eingabe$="" THEN RETURN
340 gefunden=0
341 FOR i=1 TO Vokabelzaehler
342  FOR ii=1 TO 3
343   IF Eingabe$=Bedeutung$(i,ii)THEN
344     COLOR 2,0:PRINT:PRINT Vokabel$(i);:COLOR 1,0:PRINT " - ";:COLOR 3,
        0:PRINT Bedeutung$(i,1);
345     IF Bedeutung$(i,2)<>"" THEN COLOR 1,0:PRINT ",";:COLOR 3,0:PRINT B
        edeutung$(i,2);
346     IF Bedeutung$(i,3)<>"" THEN COLOR 1,0:PRINT ",";:COLOR 3,0:PRINT B
        edeutung$(i,3);
347     gefunden=1
348   END IF
349  NEXT ii
350 NEXT i
351 IF gefunden=0 THEN COLOR 2,0:PRINT:PRINT " Suchbegriff nicht gefunden !
    !"
352 CALL WarteTaste
353 GOTO WoerterbuchDF
354 ' ************************
355 ' * Programmende/Neustart *
356 ' ************************
357 Schulz:
358 CALL JaNeinRequester
359 IF Antwort$="n" THEN RETURN
360 CLS
361 PRINT
362 COLOR 4,0:PRINT " Auf Wiedersehen, bis demnächst ..."
363 PRINT
364 MENU RESET
365 CALL Pause (2)
366 SYSTEM
367
368 Neustart:
369 CALL JaNeinRequester
370 IF Antwort$="n" THEN RETURN
371 RUN
372 ' *************
373 ' * Directory *
374 ' *************
375 VokList:
376 CLS
377 FILES ":Vokabeln"
378 CALL WarteTaste
379 RETURN
380 ' *******************
381 ' * Unterprogramme *
382 ' * Aufruf mit GOSUB *
383 ' *******************
384 Titel:
385 SCREEN 1,640,200,3,2
386 WINDOW 1,"AMIGA - Vokabeltrainer Version 2.0",,0,1
387 WINDOW OUTPUT 1
388 PALETTE 0,0,0,0        'Schwarz
389 PALETTE 1,.73,.73,.73  'Grau
390 PALETTE 2,.93,.2,0     'Feuerwehrrot
391 PALETTE 3,0,.93,.87    'Blaugruen
392 PALETTE 4,1,1,.13      'Gelb
393 PALETTE 5,.2,.4,1      'Dunkelblau
394 PALETTE 6,.33,.87,0    'Gruen
395 DIM Vokabel$(maxindex),Bedeutung$(maxindex,3),Vokabelstatus(maxindex)
396 CLS
397 COLOR 3,0:PRINT:PRINT TAB(29)"Amiga-Vokabeltrainer"
398 LINE (220,16)-(385,16),2
399 COLOR 4,0:PRINT:PRINT TAB(33)"Version 2.0"
400 PRINT:PRINT:PRINT
401 GOSUB PullDownMenues
402 RETURN
403
404 Errors:
405 MENU STOP
406 Fehlertext$=""
407 IF ERR=53 THEN Fehlertext$="Dieser Name existiert auf der Diskette nich
    t":GOTO weiter0
408 IF ERR=57 THEN Fehlertext$="Ein-/Ausgabefehler von einem Gerät":GOTO we
    iter0
409 IF ERR=58 THEN Fehlertext$="Datei existiert schon einmal":GOTO weiter0
410 IF ERR=61 THEN Fehlertext$="Diese Diskette ist voll":GOTO weiter0
411 IF ERR=64 THEN Fehlertext$="Das ist ein ungütiger Dateiname":GOTO weite
    r0
412 IF ERR=70 THEN Fehlertext$="Auf dieser Diskette ist ein Schreibschutz":
    GOTO weiter0
413 ON ERROR GOTO 0
414 weiter0:
415 PRINT:PRINT
416 COLOR 2,0:PRINT Fehlertext$;" !!":BEEP
417 CALL Pause(4)
418 CLOSE #1
419 Dateiname$=""
420 GOSUB Hauptbildschirm
421 MENU ON
422 RESUME Uhr
423
424 FrageSprache:
425 WHILE Antwort$() "j"
426  CLS:PRINT
427  COLOR 1,0:PRINT " Bitte geben Sie ein um welche ";:COLOR 2,0:PRINT "Fr
     emdsprache";:COLOR 1,0:PRINT " es sich handelt ."
428  PRINT:PRINT  " Folgende Möglichkeiten stehen zur Verfügung :"
429  COLOR 3,0:PRINT
430  LINE (1,38)-(119,48),2,b
431  PRINT TAB(3)"Englisch":PRINT
432  LINE (1,54)-(119,64),2,b
433  PRINT TAB(3)"Französisch":PRINT
434  LINE (1,70)-(119,80),2,b
435  PRINT TAB(3)"Lateinisch":PRINT
436  LINE (1,86)-(119,96),2,b
437  PRINT TAB(3)"Spanisch":PRINT
438  LINE (1,102)-(119,112),2,b
439  PRINT TAB(3)"Italienisch":PRINT
440  Weiter=0
441  WHILE Weiter=0
442    WHILE MOUSE(0)>-1:WEND
443    x=MOUSE(1):y=MOUSE(2)
444    IF (x=>1 AND x<=119 AND y=>38 AND y<=48) THEN Sprache$="Englisch":x=
       6:Weiter=1
445    IF (x=>1 AND x<=119 AND y=>54 AND y<=64) THEN Sprache$="Französisch"
       :x=8:Weiter=1
446    IF (x=>1 AND x<=119 AND y=>70 AND y<=80) THEN Sprache$="Lateinisch":
       x=10:Weiter=1
447    IF (x=>1 AND x<=119 AND y=>86 AND y<=96) THEN Sprache$="Spanisch":x=
       12:Weiter=1
448    IF (x=>1 AND x<=119 AND y=>102 AND y<=112) THEN Sprache$="Italienisc
       h":x=14:Weiter=1
449  WEND
450  LOCATE x,3:COLOR 0,3:PRINT Sprache$
451  LOCATE x,3:COLOR 3,0:PRINT Sprache$
452  LOCATE 10,30:COLOR 4,0:PRINT "Sprache = ";:COLOR 2,0:PRINT Sprache$
453  CALL JaNeinRequester
454 WEND
455 RETURN
456
457 SpeechE:
458 IF Speech$="" THEN
459  Speech$="Englisch"
460  MENU 5,1,2
461 ELSE
462  Speech$=""
463  MENU 5,1,1
```

Listing 1. Der Vokabeltrainer für den Amiga (Fortsetzung)

```
464 END IF
465 RETURN
466
467 PullDownMenues:
468 MENU 1,0,1,"Programm"
469 MENU 1,1,1," Neustart"
470 MENU 1,2,1," Ende     "
471 MENU 2,0,0,"Abfragen"
472 MENU 2,1,1," Fremdsprache - Deutsch"
473 MENU 2,2,1," Deutsch - Fremdsprache"
474 MENU 2,3,1," Beides                "
475 MENU 3,0,1,"Wörterbuch"
476 MENU 3,1,1," Fremdsprache - Deutsch"
477 MENU 3,2,1," Deutsch - Fremdsprache"
478 MENU 4,0,1,"Vokabeln"
479 MENU 4,1,1," eingeben  "
480 MENU 4,2,0," ändern    "
481 MENU 4,3,1," laden     "
482 MENU 4,4,0," speichern "
483 MENU 4,5,1," listen    "
484 MENU 5,0,1,"Sprechen"
485 MENU 5,1,1,"  Englisch  "
486 ON MENU GOSUB Verzweigen
487 MENU ON
488 RETURN
489
490 MenuesEinrichten:
491 MENU 2,0,1
492 MENU 3,0,1
493 MENU 4,2,1
494 MENU 4,4,1
495 RETURN
496
497 MenuesSperren:
498 MENU 2,0,0
499 MENU 3,0,0
500 MENU 4,2,0
501 MENU 4,4,0
502 RETURN
503
504 Verzweigen:
505 MENU STOP
506 a=MENU(0):b=MENU(1)
507 ON a GOTO Menue1,Menue2,Menue3,Menue4,Menue5
508  Menue1:
509  ON b GOSUB Neustart,Schulz
510  GOTO Weiter
511  Menue2:
512  ON b GOSUB VokAbfrageFD,VokabfrageDF,VokAbfrage
513  GOTO Weiter
514  Menue3:
515  ON b GOSUB WoerterbuchFD,WoerterbuchDF
516  GOTO Weiter
517  Menue4:
518  ON b GOSUB VokEingabe,VokAenderung,VokLaden,VokSpeichern,VokList
519  GOTO Weiter
520  Menue5:
521  ON b GOSUB SpeechE
522 Weiter:
523 GOSUB Hauptbildschirm
524 MENU ON
525 RETURN Uhr
526
527 Hauptbildschirm:
528 CLS:PRINT
529 COLOR 3,0:PRINT TAB(23)"AMIGA - Vokabeltrainer Version 2.0"
530 LINE (170,16)-(454,16),2
531 PRINT:PRINT
532 COLOR 6,0:PRINT TAB(15)"Aktive Datei          : ";:COLOR 5,0:PRINT Datein
    ame$:PRINT
533 COLOR 6,0:PRINT TAB(15)"Anzahl der Vokabeln :"; :COLOR 5,0:PRINT Vokabe
    lzaehler:PRINT
534 COLOR 6,0:PRINT TAB(15)"Fremdsprache          : ";:COLOR 5,0:PRINT Sprach
    e$:PRINT
535 COLOR 6,0:PRINT TAB(15)"Datum                 : ";:COLOR 5,0:PRINT DATE$:
    PRINT
536 COLOR 6,0:PRINT TAB(15)"Uhrzeit               : ";:COLOR 5,0:PRINT TIME$:
    PRINT
537 COLOR 6,0:PRINT TAB(15)"Sprechen              : ";:COLOR 5,0:PRINT Speech
    $:PRINT
538 IF Vokabelzaehler=0 THEN
539  GOSUB MenuesSperren
540 ELSE
541  GOSUB MenuesEinrichten
542 END IF
543 RETURN
544 ' ******************
545 ' * Unterprogramme  *
546 ' * Aufruf mit CALL *
547 ' ******************
548
549 SUB Eingeben (Nichterlaubt$,xPos%,yPos%) STATIC
550  SHARED Eingabe$
551  Eingabe$="":EingabeEnde=0
552  LOCATE xPos%,yPos%:PRINT "_";
553  WHILE EingabeEnde=0
554   a$=""
555   WHILE a$="" OR INSTR(Nichterlaubt$,a$)<>0:a$=INKEY$:WEND
556    IF a$=CHR$(13) THEN
557     EingabeEnde=-1
558    ELSEIF (a$=CHR$(27) OR a$=CHR$(127) OR a$=CHR$(139)) AND LEN (Eing
       abe$)=0 THEN
559     Eingabe$=a$
560     EingabeEnde=-1
561     GOTO EnterEnd
562    ELSEIF a$=CHR$(8)AND LEN(Eingabe$)>0 THEN
563     Eingabe$=LEFT$(Eingabe$,LEN(Eingabe$)-1)
564     PRINT CHR$(8)
565    ELSEIF (ASC(a$)=>32 AND ASC(a$)<=127)OR(ASC(a$)=>160 AND ASC(a$)<=
       255) THEN
566     IF LEN(Eingabe$)=0 THEN LOCATE xPos%,yPos%:PRINT SPACE$(40)
567     Eingabe$=Eingabe$+a$
568    END IF
569   LOCATE xPos%,yPos%
570   PRINT Eingabe$;"_";
571  EnterEnd:
572  WEND
573  PRINT CHR$(8)
574 END SUB
575
576 SUB WarteTaste STATIC
577  a$=""
578  WHILE a$="":a$=INKEY$:WEND
579 END SUB
580
581 SUB Pause (z%) STATIC
582  FOR t=1 TO z%*1000:NEXT t
583 END SUB
584
585 SUB Fehler (Fehler) STATIC
586  CLS
587  COLOR 2,0
588  PRINT
589  PRINT Fehler;"Fehler !"
590  COLOR 1,0
591  IF Fehler=0 THEN
592   PRINT
593   PRINT " Du beherrschst die Vokabeln sehr gut . Weiter so !"
594   CALL WarteTaste
595  ELSEIF Fehler=1 OR Fehler=2 THEN
596   PRINT
597   PRINT " Das ist zwar nicht das Nonplusultra , aber schon ganz gut ."

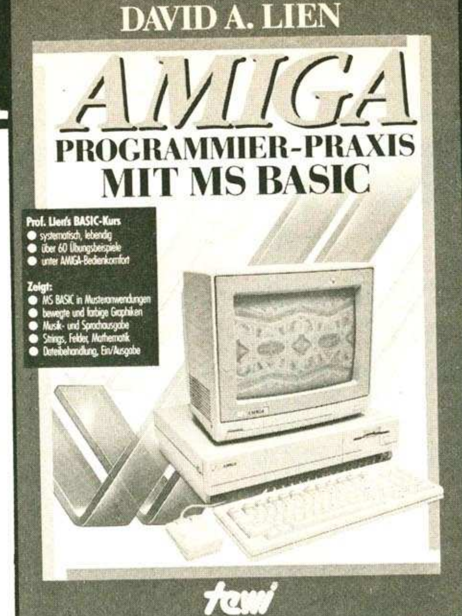
598   CALL WarteTaste
599  ELSE
600   PRINT
601   PRINT " Das Fehlermaximum ist 2 !! Also nochmal ..."
602   CALL WarteTaste
603  END IF
604 END SUB
605
606 SUB ClearVokStatus (Vokabelzaehler) STATIC
607  SHARED Vokabelstatus()
608  FOR i=1 TO Vokabelzaehler
609   Vokabelstatus(i)=0
610  NEXT i
611 END SUB
612
613 SUB Center (Text$) STATIC
614  PRINT SPC(INT((80-LEN(Text$))/2));Text$
615 END SUB
616
617 SUB JaNeinRequester STATIC
618  SHARED Antwort$,b%()
619  Antwort$="":a=0:b=0
620  GET (5,0)-(150,40),b%
621  LINE (5,0)-(150,40),1,bf
622  LINE (5,0)-(150,40),4,b
623  LINE (6,1)-(149,39),5,b
624  LINE (15,20)-(55,35),5,bf
625  LINE (14,19)-(56,36),4,b
626  LINE (100,20)-(140,35),5,bf
627  LINE (99,19)-(141,36),4,b
628  LOCATE 2,7:COLOR 2,1:PRINT "sicher ?"
629  LOCATE 4,4:COLOR 4,5:PRINT  "ja"
630  LOCATE 4,14:PRINT "nein":COLOR 1,0
631  WHILE NOT((a=>14 AND a<=56 AND b=>19 AND b<=36) OR (a=>99 AND a<=141
     AND b=>19 AND b<=36))
632   WHILE MOUSE(0)>-1:WEND
633   a=MOUSE(1):b=MOUSE(2)
634  WEND
635  IF a=>14 AND a<=56 AND b=>19 AND b<=36 THEN
636   Antwort$="j"
637  ELSE
638   Antwort$="n"
639  END IF
640  PUT (5,0),b%,PSET
641 END SUB

(c) M&T
```

Listing 1. Der Vokabeltrainer für den Amiga (Schluß)

AMIGA

MS BASIC



AMIGA bietet den Programmierkomfort!
Prof. D. Lien hält BASIC-Kurse in San Diego. Systematisch in seiner Stoffdarstellung, amerikanisch-locker in seiner Sprache, zeigt er MS BASIC unter der komfortablen Maus/Fenster Oberfläche des AMIGA.
hemen: BASIC in Musteranwendungen; Strings; Mathematik; Felder; bewegte/farbige Graphiken; Musik- und Sprachausgabe; Dateibehandlung; Ein/Ausgabe usw.

Mit über 60 Übungsbeispielen ideal für Kurse und Selbststudium.

David A. Lien, 400 Seiten, Softcover, DM 59,–

MS-BASIC bietet den Sprachkomfort!
Ausdrucksstark: 210 BASIC-Begriffe. **Ökonomisch:** Programmsynthese aus Moduln durch lokale Variablen/Wertübergaben mit COMMON/Nachladen von Segmenten mit Parameterübergabe. **Übersichtlich:** unnumerierte Zahlen für strukturierte Programmierung. **Modern:** ereignissteuerbare Programme/Gleitkommaarithmetik/Fremddateizugriffe/Kommunikationsbefehle ... usw.

Umfragen von 1986 zeigen BASIC ungebrochen als Sprachfavoriten.

HI-TEC-COMPUTER – HI-TEC-SPRACHE

tewi
tewi Verlag GmbH
Theo-Prosel-Weg 1
8000 München 40

Weitere te-wi-Bücher



M68000 FAMILIE, 2 Bd.
Hilf/Nausch, ges. 968 Seiten
Einzige Motorola-authentische Darstellung von CPU-68000-Architektur, Programmierung, Systemaufbauten. Behandelt alle 68000-Bausteine sowie 68020, 68881.
Bd. 1, Grundlagen + Architektur, 568 Seiten, DM 79,–
Bd. 2, Anwendung und Bausteine, 400 Seiten, DM 69,–



DAS C-BUCH **NEU**
Textbuch für C-Kurse und C-Anwendungen auf PCs. Beschreibt sämtliche Konstrukte der C-Sprache unter den Betriebssystemen MS DOS, CP/M, ISIS, UNIX und für die C-Compiler von MS, DR, LATTICE, INTEL. Didaktische und typographisch außergewöhnlich. Mit über 100 lauffähigen Beispielprogrammen für PCs. Zeigt Realisierungen neuester Softwarestrategien in „C".
Von Herold/Unger.
576 Seiten, Softcover. DM 79,–



LOGO –
Jeder kann programmieren
(Daniel Watt)
Buch des Jahres in den USA.
Best-rezensiert von Pädagogen und deutschen Kultusministerien. Ein bildreicher Führer durch Gedankenexperimente in LOGO.
Von Papert's Schüler D. Watt.
384 Seiten, A4, DM 59,–



UMWELTDYNAMIK
30 Programme für kybernetische Umwelterfahrungen auf allen BASIC-Rechnern. Das Buch enthält beides: Ein Programmsystem zur Simulation eigener Problemformulierungen und 29 kommentierte Modellbeispiele wie Baumsterben, Heizungsbedarf, Nahrungsketten usw. Prospekt anfordern.
Von Hartmut Bossel, 480 Seiten, Softcover, DM 59,–



BASIC Programmierung PC-10/PC-20
Durch seine Systematik ideal als Kursunterlage für PC-10/PC-20 und Kompatible. Mit Einführung in das PC-10 System und Tastendarstellung im Text. BASIC-Befehlsbeschreibungen mit Aufgaben und Antworten. Beispielprogramme. Von David Lien. 488 Seiten, Softcover. DM 59,–



C-64/SX-64 Computer-Handbuch
Die C-64 Enzyklopädie
Kompetenz durch Einsicht und Faktenwissen: einzigartige 700-Seiten-Analyse des 64er für Referenz, Ausbildung und Anwendung. Über 300 Programmierungen aller 64er Funktionen beantworten auch komplexe System/Programmfragen. Von Raeto West. 688 Seiten, Softcover, DM 66,–

tm 4712

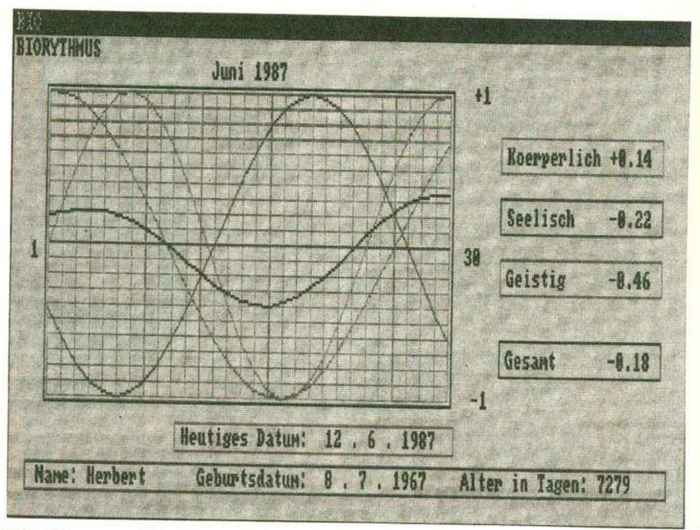
Biorhythmus mit Extras

Es gibt zwar schon viele Biorhythmusprogramme, aber die vorliegende Umsetzung für den Amiga ist besonders gut gelungen. Die gute grafische Darstellung und ein paar Besonderheiten machen das Programm interessant.

Nach dem Starten des Programmes erscheint das Hauptmenü mit den sieben Menüpunkten und den Daten des letzten Programmlaufs. Als erstes sollte man das aktuelle Datum eingeben, da die folgenden Angaben dieses Datum benutzen. Als nächstes kann man sein Geburtsdatum eingeben, das zur Berechnung der verschiedenen Kurven benötigt wird. Das Programm gibt dann aus, an welchem Wochentag man geboren wurde und wieviel Tage man alt ist. Danach wird abgefragt, ob man in die Datei mit den Geburtsdaten aufgenommen werden will. Stehen die Daten schon in dieser Datei, kann man sie bei späteren Programmläufen mit dem zweiten Menüpunkt wieder laden und erspart sich die Mühe der erneuten Eingabe. In dieser Datei können auch mehrere Personen stehen, so daß das Programm nicht nur für einen Einzelnen interessant ist.

Hat man all diese Eingaben erledigt, kann man sich die Biorhythmuskurve ansehen. Außer der gut gelungenen grafischen Darstellung der Kurven für Körper, Seele und Geist wird auch die Gesamtkurve gezeichnet. Zusätzlich werden die Werte für das aktuelle Datum noch zahlenmäßig dargestellt. All dies geschieht in recht kurzer Zeit, was angenehm auffällt. Nach einem Druck auf eine beliebige Taste gelangt man wieder ins Hauptmenü.

Was bei vielen Biorhythmusprogrammen fehlt, ist die Darstellung der Gesamtkurve für das laufende Jahr. Aber auch hier kann das Programm überzeugen. Nachdem man den fünften Menüpunkt ausgewählt hat, wird die Kurve gezeichnet. Gut gemacht ist auch die farbige Unterlegung des Monats, in dem die Kurve erstellt wird.



Die Kurven für einen Monat, hier die des Autors

Mit dem Menüpunkt »Extremwerte berechnen« kann man die Tage des Monats anzeigen lassen, an denen die Kurven ihr Minimum oder Maximum erreichen.

Der letzte Punkt im Hauptmenü schließlich dient dazu das Programm zu beenden und Amiga-Basic zu verlassen.

(Herbert Schwabl/rb)

Programmname:	Bio
Computer:	A500, A1000, A2000
Sprache:	Amiga-Basic

```
Programm : bio
--------------
 7 SUB Init (n$,t,m,j,tg,mg,jg,a,Anz,Tage(),Monat$(),TagName$()) STATIC
 8   SHARED Datum$(),Namen$()
 9     SCREEN 2,640,200,4,2
10     WINDOW 2,"BIO",(0,0)-(631,186),0,2: WINDOW OUTPUT 2
11     FOR i=0 TO 9: READ r,g,b: PALETTE i,r,g,b: NEXT i
12     DATA 1,.87,.73,0,0,0,.3,.5,.9,.9,.5,.57,.9,.08,.83,.23,.77,0
13     DATA .37,.77,.9,.9,.9,.07,1,1,1,.7,.7,.7
14     DEFINT x: DIM x(256)
15     FOR i=0 TO 127: x(i)=-128: NEXT i
16     FOR i=128 TO 255: x(i)=127: NEXT i: WAVE 0,x: ERASE x
17     FOR i=1 TO 12: READ Tage (i),Monat$ (i): NEXT i
18     DATA 31,"Januar",29,"Februar",31,"März",30,"April"
19     DATA 31,"Mai",30,"Juni",31,"Juli",31,"August"
20     DATA 30,"September",31,"Oktober",30,"November",31,"Dezember"
21     FOR i=1 TO 7: READ TagName$ (i): NEXT i
22     DATA "Montag","Dienstag","Mittwoch","Donnerstag"
23     DATA "Freitag","Samstag","Sonntag"
24     OPEN ":Namen.dat" FOR INPUT AS #2
25     INPUT #2,Anz
26     FOR i=1 TO Anz
27       INPUT #2,Namen$ (i),Datum$ (i)
28     NEXT i: CLOSE #2
29     n$="Herbert": t=28: m=4: j=1987: tg=8: mg=7: jg=1967
30     CALL Alter (tg,mg,jg,t,m,j,a)
31     CALL Daten (n$,tg,mg,jg,t,m,j,a)
32   END SUB
33
34   SUB Menue STATIC
35     CALL Clr: PRINT "MENU": LINE (233,6)-(398,17),2,b
36     LOCATE 2,31: PRINT "B I O R Y T H M U S"
37     LOCATE 5,10: PRINT "1)  Geburtsdatum eingeben"
38     LOCATE 7,10: PRINT "2)  Geburtsdatum aus Datei übernehmen"
39     LOCATE 9,10: PRINT "3)  Heutiges Datum eingeben"
40     LOCATE 11,10: PRINT "4)  Biorythmus berechnen"
41     LOCATE 13,10: PRINT "5)  Jahreskurve berechnen"
42     LOCATE 15,10: PRINT "6)  Extremwerte berechnen"
43     LOCATE 18,10: PRINT "7)  Programm beenden"
44   END SUB
45
46   SUB Eingabe (n$,tg,mg,jg,a,Anz,Namen$(),Datum$()) STATIC
47     SHARED t,m,j,Anzahl,Tage(),Monat$(),TagName$()
48       CALL Clr: PRINT "EINGABE"
49       LOCATE 3,3: PRINT "Geben Sie bitte Ihren Namen ein:"
50       LOCATE 5,6: nn$=n$: INPUT n$: SOUND 1000,1
51       IF n$="" THEN n$=nn$: EXIT SUB
52       n$=LEFT$ (n$,11): LOCATE 8,3
53       PRINT "Geben Sie bitte Ihr Geburtsdatum im Format 'TT-MM-JJJJ' e
         in:"
54       Korrekt=0
55       WHILE (NOT Korrekt)
56         LOCATE 10,6: PRINT "                                        "
57         LOCATE 10,6: INPUT d$: SOUND 1000,1
58         PRINT: PRINT "                          ";
59         PRINT "                          "
60         CALL Kontrolle (d$,tg,mg,jg,Korrekt)
61       WEND
62       CALL Alter (tg,mg,jg,t,m,j,a)
63       CALL Tag (tg,mg,jg,TagName$())
64       CALL Sternzeichen (tg,mg,jg,s$)
65       CALL Clr: PRINT: PRINT  " Hallo ";n$;" !"
66       PRINT: PRINT  "    Du bist am ";TagName$ (0);", den ";
67       PRINT tg;".";Monat$ (mg);".";jg;" geboren,"
68       PRINT "    und bist am ";t".";Monat$(m);".";j;"  ";a;"Tage alt !
         "
69       PRINT: PRINT  "    Dein Sternzeichen ist: ";s$
70       IF Anz<Anzahl THEN
71         PRINT: i$=""
72         PRINT  " Willst Du in die Datei aufgenommen werden ? (J/N)"
73         WHILE (i$<>"n")
74           i$=INKEY$
75           IF i$<>"" AND i$<>"j" AND i$<>"n" THEN BEEP
76           IF i$="n" THEN SOUND 1000,1
77           IF i$="j" THEN
78             SOUND 500,1: Anz=Anz+1: i$="n"
79             Namen$ (Anz)=n$: Datum$ (Anz)=d$
80             CALL SpeicherDatei (Anz,Namen$(),Datum$())
81           END IF
82         WEND
```

```
83    END IF
84    IF Anz>=Anzahl THEN CALL Taste
85    CALL Daten (n$,tg,mg,jg,t,m,j,a)
86  END SUB
87
88  SUB Datei (n$,tg,mg,jg,a,Anz,Namen$(),Datum$()) STATIC
89    SHARED t,m,j,Anzahl,Tage(),Monat$(),TagName$()
90      CLS: PRINT "DATEI: ";Anz;"Einträge       Noch frei:";Anzahl-An
      z
91      LINE (240,13)-(405,25),2,b
92      LOCATE 3,32: PRINT  "N A M E N L I S T E"
93      PRINT: PRINT TAB (68);"Alter am"
94      PRINT "      Name                           Geburtsdatum    ";
95      PRINT "Sternzeichen      ";STR$ (t);".";STR$ (m);".";STR$ (j)
96      PRINT: s=-1
97      FOR i=1 TO Anz
98        IF INKEY$<>"" THEN SOUND 1000,1: s=NOT s
99        IF s THEN
100         CALL Nummer (LEFT$ (Datum$ (i),2),tt,Korrekt)
101         CALL Nummer (MID$ (Datum$ (i),4,2),mm,Korrekt)
102         CALL Nummer (RIGHT$ (Datum$ (i),4),jj,Korrekt)
103         CALL Alter (tt,mm,jj,t,m,j,a)
104         CALL Sternzeichen (tt,mm,jj,s$)
105       ELSE
106         s$="": a=0
107       END IF
108       PRINT "                                               ";
109       PRINT "                                               ";
110       LOCATE CSRLIN,1: PRINT USING "_ ##_:";i;
111       PRINT TAB (7);Namen$ (i);TAB (35);Datum$ (i);
112       IF s THEN
113         PRINT TAB (50);s$;TAB (68);USING "#######";a
114       ELSE
115         PRINT
116       END IF
117       IF i MOD 10=0 AND i<>Anz THEN
118         PRINT : PRINT
119         PRINT " ------   A)uswahl   N)ächste Seite    -------";
120         Eingabe3:
121           i$=INKEY$
122           IF i$<>"" AND i$<>"a" AND i$<>"n" THEN BEEP
123           IF i$="a" THEN SOUND 500,1: GOTO Auswahl
124         IF i$<>"n" THEN Eingabe3
125         SOUND 1000,1: LINE (8,50)-(600,180),0,bf: LOCATE 8,1
126       END IF
127     NEXT i
128     Auswahl:
129       LOCATE 22,4
130       PRINT  "Bitte Nummer der gewünschten Person eingeben: ";
131       INPUT n: SOUND 1000,1: n=INT (n)
132       IF n<1 OR n>Anz THEN n=1
133       n$=LEFT$ (Namen$ (n),12)
134       CALL Nummer (LEFT$ (Datum$ (n),2),tg,Korrekt)
135       CALL Nummer (MID$ (Datum$ (n),4,2),mg,Korrekt)
136       CALL Nummer (RIGHT$ (Datum$ (n),4),jg,Korrekt)
137       CALL Alter (tg,mg,jg,t,m,j,a)
138       CLS: CALL Daten (n$,tg,mg,jg,t,m,j,a)
139     END SUB
140
141     SUB EingabeDatum (t,m,j,a) STATIC
142       SHARED n$,tg,mg,jg,Tage(),Monat$(),TagName$()
143         CALL Clr: PRINT "EINGABE": LOCATE 3,3
144         PRINT "Geben Sie bitte das heutige Datum im Format 'TT-MM-
          JJJJ' ein:"
145         Korrekt=0: dd$=d$
146         WHILE (NOT Korrekt)
147           LOCATE 5,6: PRINT "
              "
148           LOCATE 5,6: INPUT d$:SOUND 1000,1
149           IF d$="" THEN dd$=d$: EXIT SUB
150           PRINT: PRINT "
                           "
151           CALL Kontrolle (d$,t,m,j,Korrekt)
152         WEND
153         CALL Alter (tg,mg,jg,t,m,j,a)
154         CALL Tag (t,m,j,TagName$()): CALL Clr
155         PRINT: PRINT  "  Heute ist also ";TagName$ (0);", der ";
156         PRINT t;".";Monat$ (m);".";j;" ."
157         PRINT "   ";n$;" ist heute genau ";a;" Tage alt !"
158         CALL Taste: CALL Daten (n$,tg,mg,jg,t,m,j,a)
159       END SUB
160
161       SUB Biorythmus (tg,mg,jg,t,m,j) STATIC
162         SHARED Tage()
163           CALL BioGrafik (t,m,j)
164           CALL Alter (tg,mg,jg,1!,m,j,a)
165           pi=3.14151265#
166           pp=60*SIN ((a-23*INT (a/23))/23*2*pi)
167           ee=60*SIN ((a-28*INT (a/28))/28*2*pi)
168           gg=60*SIN ((a-33*INT (a/33))/33*2*pi)
169           ss=INT ((pp+ee+gg)/3)
170           FOR i=1 TO Tage (m)-1
171             p=60*SIN ((a+i-23*INT ((a+i)/23))/23*2*pi)
172             e=60*SIN ((a+i-28*INT ((a+i)/28))/28*2*pi)
173             g=60*SIN ((a+i-33*INT ((a+i)/33))/33*2*pi)
174             s=INT ((p+e+g)/3)
175             IF a+i>0 THEN
176               LINE (20+i*13,80-pp)-(33+i*13,80-p),3
177               LINE (20+i*13,80-gg)-(33+i*13,80-g),6
178               LINE (20+i*13,80-ee)-(33+i*13,80-e),5
179               LINE (20+i*13,80-ss)-(33+i*13,80-s),1
180               pp=p: ee=e: gg=g: ss=s
181             ELSE
182               pp=0: gg=0: ee=0: ss=0
183             END IF
184           NEXT i
185           CALL Alter (tg,mg,jg,t,m,j,a)
186           p=SIN ((a-23*INT (a/23))/23*2*pi)
187           e=SIN ((a-28*INT (a/28))/28*2*pi)
188           g=SIN ((a-33*INT (a/33))/33*2*pi)
189           s=(p+e+g)/3
190           LOCATE 6,71: PRINT USING "+#.##";p
191           LOCATE 9,71: PRINT USING "+#.##";e
192           LOCATE 12,71: PRINT USING "+#.##";g
193           LOCATE 16,71: PRINT USING "+#.##";s
194           CALL Taste
195         END SUB
196
197         SUB Extreme (n$,tg,mg,jg,t,m,j) STATIC
198           CALL ExtremGrafik (n$,m,j)
199           CALL Alter (tg,mg,jg,1!,m,j,a)
200           CALL Kritisch (4!,23!,a)
201           CALL Kritisch (10!,28!,a)
202           CALL Kritisch (16!,33!,a)
203           CALL Taste: CLS
204           CALL Daten (n$,tg,mg,jg,t,m,j,a)
205         END SUB
206
207         SUB Jahreskurve (tg,mg,jg,j,m) STATIC
208           CALL JahresGrafik (j,m)
209           CALL Alter (tg,mg,jg,1!,1!,j,a)
210           pi=3.14151265#
211           pp=60*SIN ((a-23*INT (a/23))/23*2*pi)
212           ee=60*SIN ((a-28*INT (a/28))/28*2*pi)
213           gg=60*SIN ((a-33*INT (a/33))/33*2*pi)
214           ss=INT ((pp+ee+gg)/3)
215           FOR i=1 TO 365
216             p=60*SIN ((a+i-23*INT ((a+i)/23))/23*2*pi)
217             e=60*SIN ((a+i-28*INT ((a+i)/28))/28*2*pi)
218             g=60*SIN ((a+i-33*INT ((a+i)/33))/33*2*pi)
219             s=INT ((p+e+g)/3)
220             IF a+i>0 THEN
221               LINE (70+i*479/365,80-ss)-(70+(i+1)*479/365,80-s),1
222               ss=s
223             ELSE
224               ss=0
225             END IF
226           NEXT i
227           CALL Taste
228         END SUB
229
230         SUB Kritisch (l,n,a) STATIC
231           SHARED m,j,Tage()
232             pi=3.14159265#: p=l: k=l
233             FOR i=0 TO Tage (m)-1
234               s=SIN ((a+i-n*INT ((a+i)/n))/n*2*pi)
235               IF a+i>=0 THEN
236                 IF s>.98 AND ss<.98 THEN
237                   LOCATE 1,10: l=l+1
238                   PRINT USING "##_.##_.####";i+1;m;j
239                 END IF
240                 IF s<-.98 AND ss>-.98 THEN
241                   LOCATE p,62: p=p+1
242                   PRINT USING "##_.##_.####";i+1;m;j
243                 END IF
244                 IF ABS (s)<.15 AND ABS (ss)>.15 THEN
245                   LOCATE k,36: k=k+1
246                   PRINT USING "##_.##_.####";i+1;m;j
247                 END IF
248               END IF
249               ss=s
250             NEXT i
251           END SUB
252
253           SUB ExtremGrafik (n$,m,j) STATIC
254             SHARED Monat$()
255               LINE (0,0)-(600,165),0,bf
256               LOCATE 1,1: PRINT "EXTREMWERTE  im  ";Monat$ (m);j
257               LINE (13,10)-(616,161),1,b
258               LINE (213,11)-(213,160),9
259               LINE (426,11)-(426,160),9
260               LOCATE 20,10: PRINT "Aktive Tage"
261               LOCATE 20,33: PRINT "Kritische Tage"
262               LOCATE 20,60: PRINT "Passive Tage"
263               LINE (16,12)-(613,52),3,b
264               LOCATE 3,4: PRINT "Köperlich"
265               LINE (16,60)-(613,100),5,b
266               LOCATE 9,4: PRINT "Seelisch"
267               LINE (16,108)-(613,148),6,b
```

Listing 1. Das Amiga-Basic-Programm von »Bio«

```
268       LOCATE 15,4: PRINT "Geistig"
269     END SUB
270
271     SUB BioGrafik (t,m,j) STATIC
272       SHARED Monat$(),Tage()
273         CALL Clr: PRINT "BIORYTHMUS"
274         LOCATE 2,24: PRINT Monat$ (m);j
275         LOCATE 3,55: PRINT "+1"
276         LOCATE 18,55: PRINT "-1"
277         LOCATE 11,3: PRINT "1"
278         LOCATE 11,23+Tage (m): PRINT STR$ (Tage (m))
279         IF m=2 AND j MOD 4=0 THEN LOCATE 11,51: PRINT STR$
          (28);" "
280         xm=Tage (m)*13+20
281         FOR i=33 TO xm STEP 13
282           LINE (i,20)-(i,140),9
283         NEXT i
284         FOR i=20 TO 140 STEP 6
285           LINE (33,i)-(xm,i),9
286         NEXT i
287         LINE (30,18)-(xm+3,142),1,b
288         LINE (33,80)-(xm,80),2
289         LINE (20+13*t,20)-(20+13*t,140),4
290         LINE (458,36)-(610,50),3,b
291         LOCATE 6,59: PRINT "Körperlich"
292         LINE (458,60)-(610,74),5,b
293         LOCATE 9,59: PRINT "Seelisch"
294         LINE (458,84)-(610,98),6,b
295         LOCATE 12,59: PRINT "Geistig"
296         LINE (458,116)-(610,130),1,b
297         LOCATE 16,59: PRINT "Gesamt"
298       END SUB
299
300       SUB JahresGrafik (j,m) STATIC
301         CALL Clr: PRINT "BIORYTHMUS"
302         LOCATE 2,37: PRINT j
303         LOCATE 3,72: PRINT "+1"
304         LOCATE 18,72: PRINT "-1"
305         LOCATE 11,5: PRINT "Jan"
306         LOCATE 11,72: PRINT "Dez"
307         LINE (30+m*40,20)-(70+m*40,140),4,bf
308         FOR i=70 TO 550 STEP 40
309           LINE (i,20)-(i,140),9
310         NEXT i
311         FOR i=20 TO 140 STEP 6
312           LINE (70,i)-(550,i),9
313         NEXT i
314         LINE (67,18)-(553,142),1,b
315         LINE (70,80)-(550,80),2
316       END SUB
317
318       SUB SpeicherDatei (Anz,Namen$(),Datum$()) STATIC
319         OPEN ":Namen.dat" FOR OUTPUT AS #2
320           PRINT# 2,Anz
321           FOR i=1 TO Anz
322             WRITE #2,Namen$ (i),Datum$ (i)
323           NEXT i
324         CLOSE# 2
325       END SUB
326
327       SUB Kontrolle (d$,tt,mm,jj,Korrekt) STATIC
328         SHARED Tage()
329           LOCATE CSRLIN-1,1: Korrekt=-1
330           IF LEN (d$)<>10 THEN Korrekt=0
331           CALL Nummer (LEFT$ (d$,2),tt,Korrekt)
332           CALL Nummer (MID$ (d$,4,2),mm,Korrekt)
333           CALL Nummer (RIGHT$ (d$,4),jj,Korrekt)
334           IF NOT Korrekt THEN
335             BEEP: PRINT "Fehlerhafte Eingabe !"
336             EXIT SUB
337           END IF
338           IF jj<0 OR jj>3000 THEN
339             BEEP: Korrekt=0
340             PRINT "Bitte nur Jahreszahlen zwischen 0000 un
            d 3000 eingeben !"
341             EXIT SUB
342           END IF
343           IF mm<1 OR mm>12 THEN
344             BEEP: PRINT  "Fehlerhaftes Monat !": Korrekt=0
345             EXIT SUB
346           END IF
347           IF tt<1 OR tt>Tage (mm) THEN
348             BEEP: PRINT "Fehlerhafter Tag !    ": Korrekt=0
349           END IF
350           IF mm=2 AND jj MOD 4<>0 AND tt>28 THEN
351             BEEP: PRINT "Fehlerhafter Tag !    ": Korrekt=0
352           END IF
353         END SUB
354
355         SUB Nummer (i$,n,Korrekt) STATIC
356           n=0
357           FOR i=1 TO LEN (i$)
358             IF MID$ (i$,i,1)>"9" OR MID$ (i$,i,1)<"0" THEN

359               Korrekt=0
360               EXIT SUB
361             END IF
362             n=10*n+ASC (MID$ (i$,i,1))-48
363           NEXT i
364         END SUB
365
366         SUB Alter (tg,mg,jg,t,m,j,a) STATIC
367           a=INT (365.25*j)-INT (j/100)+INT (j/400)+INT (30
          .6001*(m+1))+t
368           b=INT (365.25*jg)-INT (jg/100)+INT (jg/400)+INT
          (30.6001*(mg+1))+tg
369           a=a-b
370         END SUB
371
372         SUB Tag (tt,mm,jj,TagName$()) STATIC
373           CALL Alter (tt,mm,jj,0!,0!,0!,n)
374           n=ABS (n): n=n-7*INT (n/7)+1
375           TagName$ (0)=TagName$ (n)
376         END SUB
377
378         SUB Sternzeichen (tt,mm,jj,s$) STATIC
379           IF mm= 1 AND tt>=21 OR mm= 2 AND tt<=19 THEN s$=
          "Wassermann"
380           IF mm= 2 AND tt>=20 OR mm= 3 AND tt<=20 THEN s$=
          "Fische"
381           IF mm= 3 AND tt>=21 OR mm= 4 AND tt<=20 THEN s$=
          "Widder"
382           IF mm= 4 AND tt>=21 OR mm= 5 AND tt<=20 THEN s$=
          "Stier"
383           IF mm= 5 AND tt>=21 OR mm= 6 AND tt<=21 THEN s$=
          "Zwillinge"
384           IF mm= 6 AND tt>=22 OR mm= 7 AND tt<=22 THEN s$=
          "Krebs"
385           IF mm= 7 AND tt>=23 OR mm= 8 AND tt<=23 THEN s$=
          "Löwe"
386           IF mm= 8 AND tt>=24 OR mm= 9 AND tt<=23 THEN s$=
          "Jungfrau"
387           IF mm= 9 AND tt>=24 OR mm=10 AND tt<=23 THEN s$=
          "Waage"
388           IF mm=10 AND tt>=24 OR mm=11 AND tt<=22 THEN s$=
          "Skorpion"
389           IF mm=11 AND tt>=23 OR mm=12 AND tt<=21 THEN s$=
          "Schütze"
390           IF mm=12 AND tt>=22 OR mm= 1 AND tt<=20 THEN s$=
          "Steinbock"
391         END SUB
392
393         SUB Clr STATIC
394           LINE (0,0)-(617,145),0,bf: LOCATE 1,1
395         END SUB
396
397         SUB Taste STATIC
398           WHILE (INKEY$="")
399           WEND
400           SOUND 1000,1
401         END SUB
402
403         SUB Daten (n$,tg,mg,jg,t,m,j,a) STATIC
404           LINE (154,150)-(416,161),0,bf
405           LINE (154,150)-(416,161),3,b
406           LOCATE 20,21: PRINT "Heutiges Datum: ";t;".";m;"
          .";j
407           LINE (13,166)-(616,177),0,bf
408           LINE (13,166)-(616,177),5,b
409           LOCATE 22,4: PRINT "Name: ";n$;TAB (23);"Geburts
          datum: ";
410           PRINT tg;".";mg;".";jg
411           LOCATE 22,54: PRINT "Alter in Tagen:";a
412         END SUB
413
414 BEGIN:
415   Anzahl=100
416   DIM Tage (12), Monat$ (12), TagName$ (7)
417   DIM Namen$ (Anzahl), Datum$ (Anzahl)
418   ON ERROR GOTO DateiAnlegen
419   OPEN ":Namen.dat" FOR INPUT AS #2
420   CLOSE #2: GOTO Weiter
421 DateiAnlegen:
422   Anz=1: Namen$ (1)="Herbert": Datum$ (1)="08-07-1967"
423   CALL SpeicherDatei (Anz,Namen$(),Datum$())
424 Weiter: CALL Init (n$,t,m,j,tg,mg,jg,a,Anz,Tage(),Monat$(),TagName$())
425   CALL Menue
426   WHILE (i$<>"7")
427     i$=INKEY$: IF i$>="1" AND i$<="6" THEN SOUND 500,1
428     IF i$="1" THEN CALL Eingabe (n$,tg,mg,jg,a,Anz,Namen$(),Datum$())
429     IF i$="2" THEN CALL Datei (n$,tg,mg,jg,a,Anz,Namen$(),Datum$())
430     IF i$="3" THEN CALL EingabeDatum (t,m,j,a)
431     IF i$="4" THEN CALL Biorythmus (tg,mg,jg,t,m,j)
432     IF i$="6" THEN CALL Extreme (n$,tg,mg,jg,t,m,j)
433     IF i$="5" THEN CALL Jahreskurve (tg,mg,jg,j,m)
434     IF i$>="1" AND i$<="6" THEN CALL Menue
435     IF (i$<"1" OR i$>"7") AND i$<>"" THEN BEEP
436   WEND
437   WINDOW CLOSE 2: SCREEN CLOSE 2
438 END
439
(c) M&T
```

Listing 1. Das Programm »Bio« bitte ohne Zeilennummern eingeben

Vier gewinnt

Das Spiel »Vier gewinnt« ist ja hinlänglich bekannt. Die Umsetzung auf den Amiga wird mit der Maus gesteuert und regt zum Experimentieren an.

Das Spiel wird auf einem Feld von 7 waagrecht mal 6 senkrecht gespielt. Ziel ist es dabei vier Scheiben, egal ob waagrecht, senkrecht oder diagonal in eine Reihe zu bringen. Das Problem ist, daß die Scheiben immer nur von oben auf die anderen gelegt werden dürfen. Dies geschieht, indem man den Mauszeiger auf einen der sieben blauen Pfeile bewegt und die rechte Maustaste drückt. Nehmen zwei Spieler teil, wird dies immer abwechselnd getan. Die Namen der Mitspieler werden am Anfang des Programms eingegeben. Der erste Spieler nimmt die roten, der zweite die gelben Scheiben. Drückt man bei der Eingabe des Namens nur <RETURN>, übernimmt der Computer diesen Spieler. Gibt man beide Male nur <RETURN> ein, spielt der Computer gegen sich selbst, wobei dies leider nur zufallsgesteuert geschieht. Vielleicht schreibt jemand von Ihnen einen Programmteil, der hier eine bessere Spielweise des Computers ermöglicht.

Hat einer der beiden Mitspieler vier Scheiben in eine Reihe gebracht, wird dies mit einem blinkenden und piepsenden Pfeil angezeigt. Danach startet das Programm zu einer neuen Partie.

(Erik Spangenberg/rb)

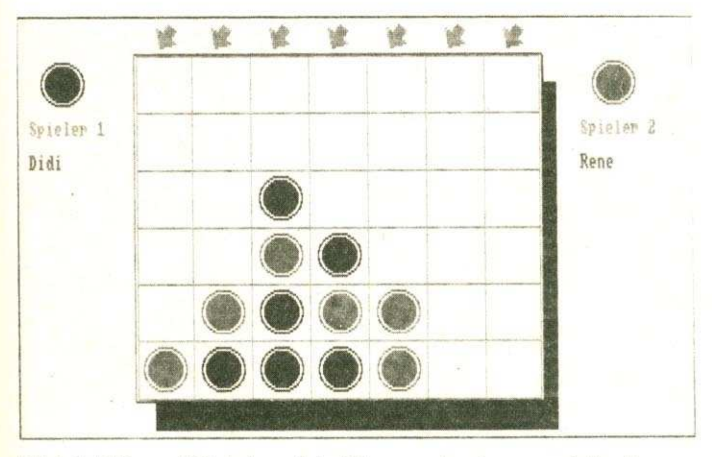


Bild 1. Mit zwei Spielern ist »Vier gewinnt« am schönsten

Programmname:	Vier gewinnt
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	Amiga-Basic
Bemerkung:	80-Zeichenmodus nötig

```
Programm : VierGewinnt
----------------------
1 '          Vier gewinnt
2 '       Erik Spangenberg
3 ' Bitte 80-Zeichen aktivieren!
4 '
5 'Versuchen Sie,eine Folge bestehend aus 4
6 'Talern in der diagonalen,senkrechten oder
7 'waagerechten Achse aufzubauen.
```

Listing 1. Das Listing zu »Vier gewinnt« (bitte ohne Zeilennummern eingeben)

```
8 '
9 SCREEN 1,640,200,4,2:WINDOW 2,"",,0,1
10 LOCATE 10,30:PRINT"Moment bitte...."
11 start:
12 CLEAR ,40000&
13 DIM a(30),b(30),c(30),d(30),a%(255),b%(255)
14 DIM xwert(50),ywert(50),spalte(50),zeile(50)
15 DIM t(110),l(110),xa(600),xb(600),xpos(50)
16 DIM ypos(50),ba(50),bb(50),pfeil%(255)
17 DIM welle1%(255),welle2%(255),leer%(255)
18 FOR w=0 TO 227:welle1%(w)=127:welle2%(w)=-127
19 NEXT w
20 WAVE 2,welle1%:WAVE 3,welle2%
21 GOSUB daten
22 CLS
23 PALETTE 15,0,.93,.87:PALETTE 14,.33,.87,0
24 PALETTE 1,.4,.4,.4:PALETTE 0,.3,.3,.35
25 PALETTE 6,.7,.7,.7:PALETTE 10,.9,.2,0
26 PALETTE 11,.33,1,0:PALETTE 12,.1,.4,1
27 PALETTE 13,1,.59,0:PALETTE 14,1,1,.13
28 PALETTE 15,0,.93,.87:PALETTE 14,.33,.87,0
29 PALETTE 5,.25,.25,.28:PALETTE 4,.33,.87,0
30 PALETTE 3,.32,.32,1:PALETTE 2,1,1,.7
31 PALETTE 9,.33,.77,.8:PALETTE 7,.46,.46,.46
32 LINE(107,16)-(491,16),1:LINE(492,16)-(492,192),1
33 LINE(107,17)-(107,179),1:LINE(492,180)-(107,180),1
34 RANDOMIZE TIMER:COLOR 15,0
35 x=60:y=20:FOR z1=1 TO 6:LINE(x+50,y-3)-(490,y-3),6
36 FOR z=1 TO 7:x=x+55:xw=xw+1:spalte(xw)=x
37 LINE(x-8,17)-(x-8,179),6:NEXT z:yw=yw+1:zeile(yw)=y
38 x=60:y=y+27:NEXT z1:LINE(x+50,y-3)-(490,y-3),6
39 LINE(107,17)-(107,179),1
40 LINE(490,17)-(490,179),6:LINE(110,17)-(110,179),6
41 LINE(493,30)-(505,30),5:LINE(505,30)-(505,193),5
42 LINE(505,193)-(126,193),5:LINE(126,193)-(126,180),5
43 LINE(126,180)-(493,180),5
44 LINE(493,180)-(493,17),5:PAINT(130,185),5
45 LINE(133,7)-(140,2),9:LINE(140,2)-(147,5),9
46 LINE(147,5)-(142,9),9
47 LINE(142,9)-(147,11),9:LINE(147,11)-(132,13),9
48 LINE(132,13)-(129,4),9
49 LINE(129,4)-(133,5),9:PAINT(135,9),9
50 GET(128,0)-(150,15),pfeil%
51 FOR j=180 TO 455 STEP 55
52 PUT(j,0),pfeil%
53 NEXT j
54 CIRCLE(40,30),20,10,,,.5:CIRCLE(40,30),16,10,,,.5
55 PAINT(45,35),10
56 CIRCLE(560,30),20,13,,,.5:CIRCLE (560,30),16,13,,,.5
57 PAINT(565,35),13
58 GET(20,20)-(60,40),a%:GET(540,20)-(580,40),b%
59 COLOR 15,0
60 anfang:
61 PALETTE 15,0,.93,.87:PALETTE 14,.33,.87,0
62 COLOR 15,0
63 LOCATE 7,2:SOUND 500,.5,255,0:PRINT"Spieler 1"
64 LOCATE 7,67:SOUND 700,.5,255,0:PRINT"Spieler 2"
65 LOCATE 9,2:SOUND 800,.5,255,0:PRINT"Name":LOCATE 9,67
66 SOUND 400,.5,255,0
67 PRINT"Name"
68 LOCATE 11,2:LINE INPUT nam1$
69 LOCATE 9,2:PRINT SPACE$(10):LOCATE 11,2:PRINT SPACE$(10)
70 COLOR 14,0
71 IF nam1$="" THEN nam1$="Computer"
72 LOCATE 9,2:SOUND 700,.5,255,0:PRINT nam1$
73 LOCATE 11,67:LINE INPUT nam2$
74 LOCATE 9,67:PRINT SPACE$(10):LOCATE 11,67
75 PRINT SPACE$(10):IF nam2$="" THEN nam2$="Computer"
76 LOCATE 9,67:SOUND 500,.5,255,0:PRINT nam2$
77 IF nam1$="Computer" AND nam2$="Computer" THEN GOTO compu
78 IF nam1$<>"Computer" AND nam2$><"Computer" THEN GO-
   TO maus
79 IF nam1$="Computer" AND nam2$<>"Computer" THEN
   GOTO compu1
80 IF nam1$<>"Computer" AND nam2$="Computer" THEN
   GOTO compu2
81 maus:
82 IF player=1 THEN
83 WHILE MOUSE(0)=0
84 WEND
85 er:
86 xm=MOUSE(1):ym=MOUSE(2)
87 FOR x1=1 TO 7
88 IF xm>spalte(x1)+5 THEN
89 IF xm<spalte(x1)+40 AND ym<20 THEN
90 sp=spalte(x1)
91 END IF
92 END IF
93 NEXT x1
94 IF sp>=115 AND sp<=445 THEN
95 SOUND sp+ze,2,255,2
96 GOSUB fallen:GOSUB oben
97 IF xa(sp+ze)=1 OR xb(sp+ze)=1 THEN
98 SOUND sp+ze,3,255,2
99 GOTO maus
100 END IF
101 ELSE
102 GOTO er
103 END IF
104 geh$="maus"
105 zu=1
106 FOR yfall=0 TO ze STEP 10
107 SOUND 1000-sp-yfall*2,1,80,1
108 PUT(sp,yfall),a%,XOR
109 y1=yfall
110 PUT(sp,y1),a%,XOR
111 NEXT yfall
112 PUT(sp,ze),a%,PSET:anzahla=anzahla+1
113 ba(anzahla)=sp+ze:SOUND sp+ze,.5,255,0
114 xa1=ba(anzahla)
115 xa(xa1)=1
116 END IF
117 IF player=2 THEN
118 WHILE MOUSE(0)=0
119 WEND
120 ds:
121 xm=MOUSE(1):ym=MOUSE(2)
122 FOR x1=1 TO 7
123 IF xm>spalte(x1)+5 THEN
124 IF xm<spalte(x1)+40 AND ym<20 THEN
125 sp=spalte(x1)
126 END IF
127 END IF
128 NEXT x1
129 IF sp>=115 AND sp>=445 THEN
130 SOUND sp+ze,2,255,2
131 GOSUB fallen:GOSUB oben
132 IF xa(sp+ze)=1 OR xb(sp+ze)=1 THEN
133 SOUND sp+ze,3,255,2
134 GOTO maus
135 END IF
136 ELSE
137 GOTO ds
138 END IF
139 geh$="maus"
140 FOR yfall=0 TO ze STEP 10
141 SOUND 1000-sp-yfall*2,1,80,1
142 PUT(sp,yfall),b%,XOR
143 y1=yfall
144 PUT(sp,y1),b%,XOR
145 NEXT yfall
146 PUT(sp,ze),b%,PSET:anzahlb=anzahlb+1
147 bb(anzahlb)=sp+ze:SOUND sp+ze,.5,255,0
148 xb1=bb(anzahlb)
149 xb(xb1)=1
150 END IF
151 GOSUB steuerung
152 player=player+1
153 IF player>2 THEN player=1
154 WHILE MOUSE(0)=0
155 WEND
156 GOTO maus
```

```
157 steuerung:
158 IF anzahla+anzahlb=42 THEN GOTO schluss
159 FOR p=1 TO 24
160 IF xa(a(p))=1 AND xa(b(p))=1 AND xa(c(p))=1 THEN
161 IF xa(d(p))=1 THEN
162 GOTO reihe
163 END IF
164 END IF
165 IF xb(a(p))=1 AND xb(b(p))=1 AND xb(c(p))=1 THEN
166 IF xb(d(p))=1 THEN
167 GOTO reihe
168 END IF
169 END IF
170 NEXT p
171 by=80:FOR u=1 TO 4:by=by+55
172 FOR k=0 TO 5
173 IF xa(by+k*27)=1 AND xa(by+55+k*27)=1 THEN
174 IF xa(by+110+k*27)=1 AND xa(by+165+k*27)=1 THEN
175 GOTO reihe
176 END IF
177 END IF
178 IF xb(by+k*27)=1 AND xb(by+55+k*27)=1 THEN
179 IF xb(by+110+k*27)=1 AND xb(by+165+k*27)=1 THEN
180 GOTO reihe
181 END IF
182 END IF
183 NEXT k ,u
184 bt=108:FOR u=1 TO 3:bt=bt+27
185 FOR k=0 TO 6
186 IF xa(bt+k*55)=1 AND xa(bt+27+k*55)=1 THEN
187 IF xa(bt+54+k*55)=1 AND xa(bt+81+k*55)=1 THEN
188 GOTO reihe
189 END IF
190 END IF
191 IF xb(bt+k*55)=1 AND xb(bt+27+k*55)=1 THEN
192 IF xb(bt+54+k*55)=1 AND xb(bt+81+k*55)=1 THEN
193 GOTO reihe
194 END IF
195 END IF
196 NEXT k,u
197 RETURN
198 schluss:
199 FOR f=1 TO 5:SOUND 800,1,255,0
200 FOR j=1 TO 100:NEXT j,f
201 reihe:
202 GOSUB neu
203 compu:
204 GOSUB steuerung
205 player=player+1:IF player>2 THEN player=1
206 IF player=1 THEN
207 geh$="compu"
208 d:
209 col=INT(RND(1)*7+1)
210 sp=xpos(col)
211 SOUND sp+ze,2,255,2
212 GOSUB fallen
213 GOSUB oben
214 IF xa(sp+ze)=1 OR xb(sp+ze)=1 THEN GOTO d
215 FOR yfall=0 TO ze STEP 10
216 SOUND 1000+sp-yfall*2,1,80,1
217 PUT(sp,yfall),b%,XOR
218 y1=yfall
219 PUT(sp,y1),b%,XOR
220 NEXT yfall
221 PUT(sp,ze),b%,PSET:anzahlb=anzahlb+1
222 bb(anzahlb)=sp+ze:SOUND sp+ze,.5,255,0
223 xb1=bb(anzahlb)
224 xb(xb1)=1
225 END IF
226 IF player=2 THEN
```

Listing 1. »Vier gewinnt« (Fortsetzung)

Wichtiger Hinweis für alle Kleinanzeigeninserenten:

Folgende Video- und Computerspiele sind von der Bundesprüfstelle, Bonn, indiziert:

Battlezone
Beach Head
Beach Head II
Blue Max
Castle Wolfenstein
Commando
Commando Libya Part I
Desert Fox
Eroticon
Falcon Patrol
Falcon Patrol II
Flyerfox
Friday the 13th
G.I. Joe I + II
Girls they want to have Fun
Green Beret
Hitler Diktator
Nice Demo
Paratrooper
Porno Dia Show
Protector II
Raid on Bungeling Bay
Raid over Moscow
Rambo II
River Raid
Seafox/Seawolf
Sex Games
Silent Service
Skyfox
Soldier One
Speed Racer
Stalag I
Swedish Erotica
Stroker
F 15 Strike Eagle
Tank Attack
Teachbusters
Theatre Europe
1942 Trainer

Der Verlag behält sich vor, bei Softwareangeboten indizierte Spiele ersatzlos zu streichen.

```
227 geh$="compu"
228 d1:
229 co=INT(RND(1)*7+1)
230 sp=xpos(co)
231 SOUND sp+ze,2,255,2
232 GOSUB fallen
233 GOSUB oben
234 IF xa(sp+ze)=1 OR xb(sp+ze)=1 THEN GOTO d1
235 FOR yfall=0 TO ze STEP 10
236 SOUND 1000+sp-yfall*2,1,80,1
237 PUT(sp,yfall),a%,XOR
238 y1=yfall
239 PUT(sp,y1),a%,XOR
240 NEXT yfall
241 PUT(sp,ze),a%,PSET:anzahla=anzahla+1
242 ba(anzahla)=sp+ze
243 SOUND sp+ze,.5,255,0
244 xa1=ba(anzahla)
245 xa(xa1)=1
246 END IF
247 GOTO compu
248 compu1:
249 IF player=2 THEN
250 WHILE MOUSE(0)=0
251 WEND
252 xm=MOUSE(1):ym=MOUSE(2)
253 FOR x1=1 TO 7
254 IF xm>spalte(x1)+5 THEN
255 IF xm<spalte(x1)+40 AND ym<20 THEN
256 sp=spalte(x1)
257 geh$="compu1"
258 SOUND sp+ze,2,255,2
259 GOSUB fallen:GOSUB oben
260 IF xa(sp+ze)=1 OR xb(sp+ze)=1 THEN
261 SOUND sp+ze,3,255,2
262 GOTO compu1
263 END IF
264 END IF
265 END IF
266 NEXT x1
267 FOR yfall=0 TO ze STEP 10
268 SOUND 1000+sp-yfall*2,1,80,1
269 PUT(sp,yfall),b%,XOR
270 y1=yfall
271 PUT(sp,y1),b%,XOR
272 NEXT yfall
273 PUT(sp,ze),b%,PSET:anzahlb=anzahlb+1
274 bb(anzahlb)=sp+ze:SOUND sp+ze,.5,255,0
275 xb1=bb(anzahlb)
276 xb(xb1)=1
277 END IF
278 IF player=1 THEN
279 erik:
280 co4=INT(RND(1)*7+1)
281 sp=xpos(co4)
282 geh$="compu1"
283 SOUND sp+ze,2,255,2
284 GOSUB fallen:GOSUB oben
285 IF xa(sp+ze)=1 OR xb(sp+ze)=1 THEN GOTO erik
286 FOR yfall=0 TO ze STEP 10
287 SOUND 1000+sp-yfall*2,1,80,1
288 PUT(sp,yfall),a%,XOR
289 y1=yfall
290 PUT(sp,y1),a%,XOR
291 NEXT yfall
292 PUT(sp,ze),a%,PSET:anzahla=anzahla+1
293 ba(anzahla)=xpos(co4)+ze
294 SOUND sp+ze,.5,255,0
295 xa1=ba(anzahla)
296 xa(xa1)=1
297 END IF
298 player=player+1
299 IF player>2 THEN player=1
300 GOSUB steuerung
301 WHILE MOUSE(0)=0
302 WEND
303 GOTO compu1
304 compu2:
305 IF player=1 THEN
306 WHILE MOUSE(0)=0
307 WEND
308 xm=MOUSE(1):ym=MOUSE(2)
309 FOR x1=1 TO 7
310 IF xm>spalte(x1)+5 AND xm<spalte(x1)+40 THEN
311 sp=spalte(x1)
312 geh$="compu2"
313 SOUND sp+ze,2,255,2
314 GOSUB fallen:GOSUB oben
315 IF xa(sp+ze)=1 OR xb(sp+ze)=1 THEN
316 SOUND sp+ze,2,255,3
317 GOTO compu2
318 END IF
319 END IF
320 NEXT x1
321 FOR yfall=0 TO ze STEP 10
322 SOUND 1000+sp-yfall*2,1,80,1
323 PUT(sp,yfall),a%,XOR
324 y1=yfall
325 PUT(sp,y1),a%,XOR
326 NEXT yfall
327 PUT(sp,ze),a%,PSET:anzahla=anzahla+1
328 ba(anzahla)=sp+ze
329 SOUND sp+ze,.5,255,0
330 xa1=ba(anzahla)
331 xa(xa1)=1
332 END IF
333 IF player=2 THEN
334 d3:
335 co4=INT(RND(1)*7+1)
336 sp=xpos(co4)
337 geh$="compu2"
338 SOUND sp+ze,2,255,2
339 GOSUB oben:GOSUB fallen
340 IF xa(sp+ze)=1 OR xb(sp+ze)=1 THEN GOTO d3
341 IF ze=20 AND sp+ze=1 THEN d3
342 FOR yfall=0 TO ze STEP 10
343 SOUND 1000+sp-yfall*2,1,80,1
344 PUT(sp,yfall),b%,XOR
345 y1=yfall
346 PUT(sp,y1),b%,XOR
347 NEXT yfall
348 PUT(sp,ze),b%,PSET:anzahlb=anzahlb+1
349 bb(anzahla)=sp+ze:SOUND sp+ze,.5,255,0
350 xb1=bb(anzahlb)
351 xb(xb1)=1
352 END IF
353 player=player+1
354 IF player>2 THEN player=1
355 GOSUB steuerung
356 WHILE MOUSE(0)=0
357 WEND
358 GOTO compu2
359 fallen:
360 lm:
361 IF sp=115 THEN
362 IF xb(270)=0 OR xa(270)=0 THEN ze=155
363 g=155:FOR i=1 TO 5
364 g=g-27
365 IF xa(m1(i))=0 AND xa(m2(i))=1 THEN
366 ze=g
367 END IF
368 IF xb(m1(i))=0 AND xb(m2(i))=1 THEN
369 ze=g
370 END IF
371 NEXT i
372 END IF
373 ln:
374 IF sp=170 THEN
375 IF xa(325)=0 OR xb(325)=0 THEN ze=155
376 g=155:FOR i=1 TO 5
377 g=g-27
378 IF xa(n1(i))=0 AND xa(n2(i))=1 THEN
```

```
379 ze=g
380 END IF
381 IF xb(n1(i))=0 AND xb(n2(i))=1 THEN
382 ze=g
383 END IF
384 NEXT i
385 END IF
386 lo:
387 IF sp=225 THEN
388 IF xa(380)=0 OR xb(380)=0 THEN ze=155
389 g=155:FOR i=1 TO 5
390 g=g-27
391 IF xa(o1(i))=0 AND xa(o2(i))=1 THEN
392 ze=g
393 END IF
394 IF xb(o1(i))=0 AND xb(o2(i))=1 THEN
395 ze=g
396 END IF
397 NEXT i
398 END IF
399 lp:
400 IF sp=280 THEN
401 IF xa(435)=0 OR xb(435)=0 THEN ze=155
402 g=155:FOR i=1 TO 5
403 g=g-27:IF xa(p1(i))=0 AND xa(p2(i))=1 THEN
404 ze=g
405 END IF
406 IF xb(p1(i))=0 AND xb(p2(i))=1 THEN
407 ze=g
408 END IF
409 NEXT i
410 END IF
411 lq:
412 IF sp=335 THEN
413 IF xa(490)=0 OR xb(490)=0 THEN ze=155
414 g=155:FOR i=1 TO 5
415 g=g-27
416 IF xa(q1(i))=0 AND xa(q2(i))=1 THEN
417 ze=g
418 END IF
419 IF xb(q1(i))=0 AND xb(q2(i))=1 THEN
420 ze=g
421 END IF
422 NEXT i
423 END IF
424 lr:
425 IF sp=390 THEN
426 IF xa(545)=0 OR xb(545)=0 THEN ze=155
427 g=155:FOR i=1 TO 5
428 g=g-27
429 IF xa(r1(i))=0 AND xa(r2(i))=1 THEN
430 ze=g
431 END IF
432 IF xb(r1(i))=0 AND xb(r2(i))=1 THEN
433 ze=g
434 END IF
435 NEXT i
436 END IF
437 ls:
438 IF sp=445 THEN
439 IF xa(600)=0 OR xb(600)=0 THEN ze=155
440 g=155:FOR i=1 TO 5
441 g=g-27
442 IF xa(s1(i))=0 AND xa(s2(i))=1 THEN
443 ze=g
444 END IF
445 IF xb(s1(i))=0 AND xb(s2(i))=1 THEN
446 ze=g
447 END IF
448 NEXT i
449 END IF
450 RETURN
451 oben:
452 FOR x=1 TO 7
453 IF sp=w1(x) AND ze=w2(x) AND xa(w3(x))=1 THEN
454 SOUND 500,1,255,0:GOTO geh
455 ELSE
456 RETURN
457 END IF
458 IF sp=w1(x) AND ze=w2(x) AND xb(w3(x))=1 THEN
459 SOUND 500,1,255,0:GOTO geh
460 ELSE
461 RETURN
462 END IF
463 NEXT x
464 geh:
465 IF geh$="compu" THEN GOTO compu
466 IF geh$="compu1" THEN GOTO compu1
467 IF geh$="compu2" THEN GOTO compu2
468 IF geh$="maus" THEN GOTO maus
469 IF geh$="gj" THEN GOTO gj
470 IF geh$="lk" THEN GOTO lk
471 END
472 neu:
473 GET(128,0)-(150,15),pfeil%
474 FOR mal=1 TO 20
475 PUT(sp+11,ze+3),pfeil%,XOR
476 FOR n=1 TO 200:NEXT n:SOUND 500,3,255,2
477 PUT(sp+11,ze+3),pfeil%,OR
478 FOR n=1 TO 200:NEXT n
479 NEXT mal
480 GET(20,150)-(60,170),leer%
481 PALETTE 15,.35,.35,.35:PALETTE 14,.35,.35,.35
482 LOCATE 11,2:PRINT SPACE$(10):LOCATE 11,67
483 PRINT SPACE$(10):LOCATE 9,2:PRINT SPACE$(10)
484 LOCATE 9,67:PRINT SPACE$(10)
485 xs=60:ys=20
486 FOR n1=1 TO 6:FOR n=1 TO 7
487 xs=xs+55:PUT(xs,ys),leer%,PSET
488 NEXT n:xs=60:ys=ys+27:NEXT n1
489 CLEAR:GOTO start
490 daten:
491 RESTORE yo:FOR j=1 TO 7:READ yoben(j):NEXT j
492 RESTORE iks:FOR j=1 TO 7:READ xpos(j):NEXT j
493 RESTORE datas1:FOR i=1 TO 5:READ m1(i),m2(i):NEXT i
494 RESTORE datas2:FOR i=1 TO 5:READ n1(i),n2(i):NEXT i
495 RESTORE datas3:FOR i=1 TO 5:READ o1(i),o2(i):NEXT i
496 RESTORE datas4:FOR i=1 TO 5:READ p1(i),p2(i):NEXT i
497 RESTORE datas5:FOR i=1 TO 5:READ q1(i),q2(i):NEXT i
498 RESTORE datas6:FOR i=1 TO 5:READ r1(i),r2(i):NEXT i
499 RESTORE datas7:FOR i=1 TO 5:READ s1(i),s2(i):NEXT i
500 RESTORE ob:FOR x=1 TO 7:READ w1,w2,w3:NEXT x
501 RESTORE steue:FOR n=1 TO 24:READ a(n),b(n),c(n),d(n)
502 NEXT n
503 RETURN
504 datas1: DATA 243,270,216,243,189,216,162,189,135,162
505 datas2: DATA 298,325,271,298,244,271,217,244,190,217
506 datas3: DATA 353,380,326,353,299,326,272,299,245,272
507 datas4: DATA 408,435,381,408,354,381,327,354,300,327
508 datas5: DATA 463,490,436,463,409,436,382,409,355,382
509 datas6: DATA 518,545,491,518,464,491,437,464,410,437
510 datas7: DATA 573,600,546,573,519,546,492,519,465,492
511 steue:
512 DATA 135,217,299,381,162,244,326,408,189,271,353,435
513 DATA 190,272,354,436,245,327,409,491,300,382,464,546
514 DATA 244,326,408,490,217,299,381,463,272,354,436,518
515 DATA 327,409,491,573,299,381,463,545,354,436,518,600
516 DATA 300,272,244,216,355,327,299,271,410,382,354,326
517 DATA 465,437,409,381,492,464,436,408,519,491,463,435
518 DATA 464,436,408,380,437,409,381,353,409,381,353,325
519 DATA 354,326,298,270,382,354,326,298,327,299,271,243
520 ob:
521 DATA 115,20,135,170,20,190,225,20,245,280,20,300,335,
         20,355
522 DATA 390,20,410,445,20,465
523 iks: DATA 115,170,225,280,335,390,445
524 yo: DATA 135,190,245,300,355,410,465
525 END
(c) M&T
```

Listing 1. »Vier gewinnt« (Schluß)

Hilfsmittel zur Eingabemasken-Erstellung

Wenn man ein Programm erstellt, das auch einen professionellen Eindruck machen soll, muß man mit Eingabemasken arbeiten. »Mask« ist eine Sammlung von Unterprogrammen, die das Erstellen von Masken zum Kinderspiel machen.

Durch die Verwendung der Unterroutinen von Mask (Listing 1) sind Sie in der Lage, Masken aufzubauen, bei denen Sie viele Parameter bestimmen können. Nicht nur die erlaubten Zeichen, sondern auch die Darstellung der Zeichen auf dem Bildschirm können beeinflußt werden. Außerdem können falsche Eingaben in einem Feld mit der Backspace-Taste (Taste mit dem Pfeil nach links) wieder gelöscht werden.

Mask beinhaltet zwar einige Funktionen, aber für den Benutzer sind eigentlich nur drei davon interessant. Die restlichen Funktionen werden nur von den Unterprogrammen benutzt. Die wichtigen Funktionen sollen jetzt kurz vorgestellt werden.

Die erste Funktion, die verwendet wird ist »SetField«. Mit ihr werden die Parameter für ein einzelnes Feld nach Bedarf gesetzt. Der Aufruf sieht folgendermaßen aus:

```
CALL SetField (FieldNo, menge$, länge, zeile, par2,
par3, par14567, prompt$, pz, ps)
```

Da doch eine große Zahl von Parametern übergeben werden, sollen diese jetzt einzeln erklärt werden.

»FieldNo« ist die Nummer des Feldes, für die die Parameter gelten sollen. Mit der Anweisung

```
DIM SHARED fields$ (40,14)
```

sind in der abgedruckten Version bis zu 40 Felder möglich, was sich aber leicht ändern läßt. Ein sehr wichtiger Parameter ist »menge$«, da mit ihm festgelegt wird, welche Zeichen bei der Eingabe in dem Feld überhaupt akzeptiert werden. Die Angaben müssen hier mit »"« begrenzt sein, da es sich um Stringvariablen handelt. In diesem String müssen die erlaubten Zeichen durch Kommas getrennt stehen. Nun wäre es natürlich sehr aufwendig, das gesamte Alphabet oder alle Ziffern hier hineinzuschreiben. Deshalb können das Alphabet und die Ziffern abgekürzt werden. »A-Z« steht für alle Großbuchstaben, »0-9« für alle Ziffern. Zeichen wie »-«, ».« oder die Umlaute müssen jedoch einzeln eingesetzt werden. Beispiele können Sie auch in Listing 2 sehen.

Der Parameter »länge« ist leicht zu verstehen, es handelt sich hier um die Länge des Eingabefeldes.

Die vertikale Position des Eingabefeldes wird durch »zeile« bestimmt, wobei der Wert zwischen 0 und 23 liegen muß.

»par2« und »par3« dienen zur Festlegung des linken und rechten Randes des Eingabefeldes. Es kann also auf dem Bildschirm das Feld länger sein als die Maximallänge der Eingabe.

Etwas schwieriger ist da schon »par1« zu ermitteln, da mehrere mögliche Kombinationen anwählbar sind.

Wünscht man keine Markierung, muß der Wert 0 sein. 1 und 2 stehen für die Markierung des Anfangs und des Endes. Soll an der aktuellen Eingabeposition ein Cursor erscheinen, setzt man 4 ein. Mit 8 schließlich erscheint das gesamte Eingabefeld in inverser Darstellung. Diese Werte lassen sich durch Addition auch kombinieren, so bewirkt der Wert 11 (1+2+8) ein inverses Feld mit Anfangs- und Endemarkierung.

Ob die Eingabe mit <RETURN> abgeschlossen werden muß, oder nach Eingabe von soviel Zeichen wie in Länge angegeben abgeschlossen ist, kann mittels »par4« bestimmt werden. Setzt man par4 auf 1, ist ein <RETURN> erforderlich, gibt man hingegen 0 an, wird automatisch beendet.

Sehr nützlich ist auch »par5«. Ist der Wert 1, wird der eingegebene Text automatisch groß geschrieben, bei 0 wird der Text so dargestellt, wie er eingetippt wird.

Nach der Eingabe werden die Feldbegrenzungen gelöscht, wenn für »par6« eine 1 steht, bei einer 0 bleiben die Grenzen auf dem Bildschirm.

Das Löschen des Prompts nach der Eingabe läßt sich mit »par7« beeinflussen: 0 bedeutet Löschen, bei 1 bleibt das Prompt stehen.

Die oben aufgeführten Parameter par1, par4, par5, par6 und par7 werden aber nicht einzeln übergeben, sondern erst zusammengefaßt zum Parameter »par14567«. Die Berechnung erfolgt mit der Formel

par14567 = par4+2*par5+4*par6+8*par7+16*par1

Ein Feld mit Anfangsmarkierung (par1=1), Abschluß durch <RETURN> (par4=1), ohne Großschreibung (par5=0), ohne Löschen der Feldbegrenzung (par6=0) und Löschen des Prompts (par7=1) bekommt also für par14567 den Wert 25.

Wesentlich leichter ist nun wieder der Übergabeparameter »prompt$« zu erklären. Hier handelt es sich um den Text, der erscheint, bevor die Eingabe erfolgt. Im Demoprogramm zum Beispiel die Anweisung »Vornamen eingeben«.

Die letzten zwei Variablen »pz« und »ps« geben dann noch die Zeile und Spalte an, in der das Prompt erscheinen soll. Für pz sollten die Werte zwischen 0 und 24 liegen, bei ps zwischen 0 und 79.

Nach der Ermittlung all dieser Übergabewerte steht also nun genau fest, welche Zeichen wo und wie eingegeben werden können. Der Aufruf, um nun Daten einzulesen, ist erfreulicherweise wesentlich einfacher:

```
CALL GetField (FieldNo, string)
```

Genau wie bei dem Aufruf von SetField gibt FieldNo wieder die Nummer des Feldes an, mit dem gearbeitet werden soll. Das Ergebnis des Funktionsaufrufs wird dann in »string« abgelegt. Will man numerische Werte einlesen, kann der String anschließend mit VAL(string) umgewandelt werden.

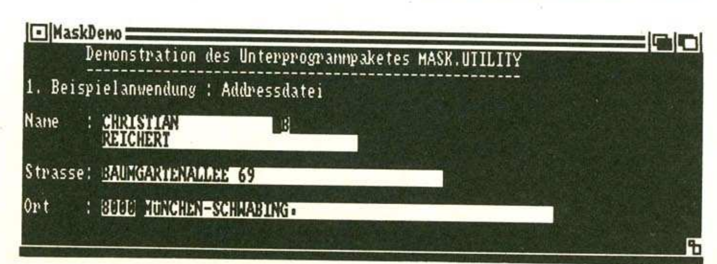
Dies waren die zwei Funktionen, die wohl hauptsächlich benutzt werden, aber es gibt noch eine Prozedur, die manchmal sehr nützlich sein kann. Will man zum Beispiel ein Feld mit einem String vorbelegen, so läßt sich dies mit SetField nicht verwirklichen. Man kann sich aber der Funktion »WriteIntoField« bedienen, die auch von GetField aufgerufen wird.
Der Aufruf lautet:

```
CALL WriteIntoField (FieldNo, text$, boundaries)
```

FieldNo übernimmt wieder dieselbe Aufgabe wie bei den anderen Funktionsaufrufen. Der Text, der im Feld erscheinen soll, muß mit »text$« übergeben werden. Ob hier eine Variable oder ein konstanter String steht, ist wie überall sonst auch vollkommen egal. Mit »boundaries« wird schließlich festgelegt, ob die Feldbegrenzungen gesetzt werden sollen oder nicht.

Durch die Kommentierung des Programms sollte es Ihnen nicht schwerfallen die anderen Funktionen von Mask zu verwenden, wenn Sie sie unbedingt benötigen. (Michael Baas/rb)

Programmname:	Mask
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	Amiga-Basic



Ein Beispielbildschirm, erzeugt mit »Mask«

```
Programm : mask
---------------

  1 '       Bedeutung der Parameter :
  2 '       menge$ ==) Definiert die Menge der Zeichen, die als
  3 '                  Eingabe zugelassen sind.
  4 '       par1   ==) Bestimmt je nach Wert die Randkennung der Maske:
  5 '                   =0 : nichts markieren   (Default)
  6 '                   =1 : Anfang markieren
  7 '                   =2 : Ende   markieren
  8 '                   =4 : die Eingabepositionen werden mit _ markiert
  9 '                   =8 : Eingabefeld wird invers dargestellt
 10 '                   Kombinationen durch Addition
 11 '       par2   ==) linker Rand des Eingabefeldes
 12 '       par3   ==) rechter Rand des Eingabefeldes
 13 '       par4   ==) Bestimmt Abbruchbedingung der Eingabe :
 14 '                   =1 : durch Return   (Default)
 15 '                   =0 : nach Eingabe von (laenge) Zeichen
 16 '       par5   ==) Eingabetext wird automatisch großgeschrieben
 17 '                   =1 : ja
 18 '                   =0 : nein  (Default)
 19 '       par6   ==) Löschen der Feldbegrenzungen nach der Eingabe
 20 '                   =1 : ja
 21 '                   =0 : nein   (Default)
 22 '       par7   ==) Löschen des Prompts nach der Eingabe
 23 '                   =1 : ja
 24 '                   =0 : nein
 25 '       laenge ==) Bestimmt die maximale Länge des Textes
 26 '       text$  ==) wird zurückgegeben, enthält die Eingabe
 27 '
 28 '       par14567 = par4 + 2 * par5 + 4 * par6 + 8 * par7 + 16 * par1
 29
 30
 31
 32
 33
 34
 35
 36 ' M A S K - U T I L I T Y
 37 ' Programm zur kontrollierten Texteingabe
 38 ' Autor     : Michael Baas
 39  - Hintergrundfarbe = 0
 40 ' - Dimensionierung des Feldes FIELDS$ (Anzahl der verschiedenen Felder
    ,14)
 41 ' - DEFINT p,l,z
 42
 43 ' DEFINT a-z
 44 DIM SHARED fields$ (40,14)
 45
 46 SUB SetField (Fieldno,menge$,laenge,zeile,par2,par3,par14567,prompt$,pz
    ,ps) STATIC
 47 ' Default-Werte setzen
 48  par4 = (par14567 AND   1)
 49  par5 = (par14567 AND   2) /  2
 50  par6 = (par14567 AND   4) /  4
 51  par7 = (par14567 AND   8) /  8
 52  par1 = (par14567 AND 240) / 16
 53
 54  IF zeile  <  1 THEN zeile  =  1
 55  IF zeile  > 24 THEN zeile  = 24
 56  IF par1   > 15 THEN par1   = 15
 57  IF par1   <  0 THEN par1   =  0
 58  IF par2   <  2 THEN par2   =  2
 59  IF par2   > 77 THEN par2   = 77
 60  IF par3   > 77 THEN par3   = 77
 61  IF par4   <  0 THEN par4   =  1
 62  IF par4   >  1 THEN par4   =  1
 63  IF par5   <  0 THEN par5   =  0
 64  IF par5   >  1 THEN par5   =  1
 65  IF par6   >  1 THEN par6   =  1
 66  IF par6   <  0 THEN par6   =  0
 67  IF par7   <  0 THEN par7   =  0
 68  IF par7   >  1 THEN par7   =  1
 69  IF par3   < par2 THEN par3 = par2 + 2
 70  IF par3   > par2 + laenge THEN par3 = par2 + laenge
 71
 72  CALL scann (menge$,menge2$)
 73  fields$ (Fieldno,0) = STR$ (zeile)
 74  fields$ (Fieldno,4) = menge2$
 75  fields$ (Fieldno,5) = STR$ (laenge)
 76  fields$ (Fieldno,6) = STR$ (par1)
 77  fields$ (Fieldno,7) = STR$ (par2)
 78  fields$ (Fieldno,8) = STR$ (par3)
 79  fields$ (Fieldno,9) = STR$ (par4)
 80  fields$ (Fieldno,10)= STR$ (par5)
 81  fields$ (Fieldno,11)= STR$ (par6)
 82  fields$ (Fieldno,12)= prompt$
 83  fields$ (Fieldno,13)= STR$ (pz)
 84  fields$ (Fieldno,14)= STR$ (ps)
 85  fields$ (Fieldno, 1)= STR$ (par7)
 86
 87  CALL ComputeEOF (Fieldno,ez,es)
 88  fields$ (Fieldno,2) = STR$ (ez)
 89  fields$ (Fieldno,3) = STR$ (es)
 90 END SUB
 91
 92 SUB ClearField (Fieldno) STATIC
 93 'Löscht den Inhalt des Feldes mit der Nummer Fieldno
 94 'Kann auch vom Benutzer aufgerufen werden
 95  ey  = VAL (fields$ (Fieldno,2)) * 8
 96  ay  = (VAL (fields$ (Fieldno,0)) - 1) * 8
 97  ax  = (VAL (fields$ (Fieldno,7)) - 1) * 8
 98  ex  = (VAL (fields$ (Fieldno,3))) * 8 - 1
 99  par3 =(VAL (fields$ (Fieldno,8))) * 8 - 1
100  LINE (ax,ay) - (par3,ey-8),0,bf
101  LINE (ax,ay) - (ex,ey)    ,0,bf
102 END SUB
```

```
103
104 SUB ComputeEOF (Fieldno,ezeile,espalt) STATIC
105 'Berechnet Endzeile und Endspalte des Feldes Fieldno
106 'Wird von dem Unterprogramm SetField aufgerufen, für
107 'den Benutzer weniger wichtig.
108  szeile = VAL (fields$ (Fieldno,0))
109  laenge = VAL (fields$ (Fieldno,5))
110  par2   = VAL (fields$ (Fieldno,7))
111  par3   = VAL (fields$ (Fieldno,8))
112  cpl    = (par3-par2) + 1
113  IF laenge <= cpl THEN
114    espalt = par2 + laenge - 1
115    ezeile = szeile
116    fields$ (Fieldno,8) = STR$ (par2 + laenge)
117  ELSE
118    onelin = laenge / cpl
119    ezeile = szeile + INT (onelin)
120    espalt = par3 + laenge - INT(onelin) * cpl
121  END IF
122 END SUB
123
124 SUB scann (menge$,zei$) STATIC
125 'Wertet den Ausdruck MENGE$, der die Menge der erlaubten Zeichen
126 'beschreibt, aus und bringt jedes einzelne erlaubte Zeichen in
127 'ZEI$, so daß eine einfache Überprüfung mittels der INSTR-Funktion mögl
    ich ist.
128 'menge$ = "0-9"
129 '        erlaubte Menge : alle Ziffern
130 'menge$ = "y,j,n,Y,J,N"
131 '        erlaubte Menge : Die Buchstaben y,j,n in Groß- und Kleinschreib
    ung
132
133  zei$ = ""
134 WHILE LEN(menge$) > 0
135  teil$ = LEFT$ (menge$,3)
136    IF INSTR (teil$,"-") = 0 THEN
137      zei$ = zei$ + LEFT$ (teil$,1)
138      cut = 2
139      GOTO endloop
140    END IF
141    IF LEFT$ (teil$,1) = "-" THEN
142      zei$ = zei$ + "-"
143      cut = 1
144      GOTO endloop
145    END IF
146    start = ASC (LEFT$(teil$,1))
147    ende  = ASC (RIGHT$(teil$,1))
148    FOR code = start TO ende
149      zei$ = zei$ + CHR$ (code)
150    NEXT code
151    cut = 4
152  endloop:
153    IF LEN(menge$) <= cut THEN menge$ = "" ELSE menge$ = RIGHT$ (menge$,
       LEN(menge$)-cut)
154 WEND
155 'Ebenfalls erlaubt : DELETE und RETURN
156 zei$ = zei$ + CHR$ (8) + CHR$ (13)
157 END SUB
158
159 SUB MarkEOF (Fieldno,farbe) STATIC
160 'Markiert das Ende des Feldes FIELDNO entsprechend der definierten
161 'Randmarkierung in der Farbe farbe (0 = lösche Rand, 1 = setze Rand)
162 azeile = VAL (fields$ (Fieldno,0))
163 aspalt = VAL (fields$ (Fieldno,7))
164 ezeile = VAL (fields$ (Fieldno,2))
165 espalt = VAL (fields$ (Fieldno,3))
166 modus  = VAL (fields$ (Fieldno,6))
167 par2   = aspalt
168 par3   = VAL (fields$ (Fieldno,8))
169 ay     = (azeile - 1) * 8
170 ax     = (aspalt - 1) * 8
171 ex     =  espalt  * 8
172 ey     = (ezeile - 1) * 8 - 1
173
174 IF (modus AND 1) > 0 THEN GOSUB startmark
175 IF (modus AND 2) > 0 THEN GOSUB endmark
176 IF (modus AND 4) > 0 THEN GOSUB underline
177 IF (modus AND 8) > 0 THEN GOSUB invers
178 EXIT SUB
179
180 startmark:
181  LINE (ax-8,ay)-(ax-4,ay+4),farbe
182  LINE -(ax-8,ay+8),farbe
183  LINE -(ax-8,ay),farbe
184 RETURN
185
186 endmark:
187  LINE (ex+8,ey)-(ex+4,ey+4),farbe
188  LINE -(ex+8,ey+8),farbe
189  LINE -(ex+8,ey),farbe
190 RETURN
191
192 underline:
193 IF (modus AND 8) > 0  THEN COLOR 0,1 ELSE COLOR 1,0
194  FOR zeile = azeile TO ezeile
195    IF zeile < ezeile THEN par31 = par3 ELSE par31 = espalt
196    FOR spalte = par2 TO par31
197      LOCATE zeile,spalte
198      PRINT "_";
199    NEXT  spalte
200  NEXT  zeile
201 RETURN
202
203 invers:
```

Listing 1. Die Unterroutinen von »Mask« (Bitte ohne Zeilennummern eingeben)

```
204  IF (modus AND 4) > 0 THEN RETURN
205  IF ey > ay + 8 THEN
206    LINE (ax,ay)-(par3 * 8 - 1,ey-8),farbe,bf
207    LINE (ax,ay)-(ex - 1,ey),farbe,bf
208  ELSE
209    LINE (ax,ay)-(ex - 1,ay + 7),farbe,bf
210  END IF
211 RETURN
212
213 END SUB
214
215 SUB WriteIntoField (Fieldno,text$,boundaries) STATIC
216 'Schreibt den Text text$ in das Feld Fieldno.
217 'Der Parameter boundaries gibt an, ob die Feldbegrenzungen
218 'gesetzt werden sollen ( = 1) oder nicht ( <> 1).
219 'Kann auch vom Benutzer aufgerufen werden.
220  IF boundaries = 1 THEN CALL MarkEOF (Fieldno)
221  IF (VAL (fields$ (Fieldno,6)) AND 8) > 0 THEN
222    COLOR 0,1
223  END IF
224  azeile = VAL (fields$ (Fieldno,0))
225  aspalt = VAL (fields$ (Fieldno,7))
226  par3   = VAL (fields$ (Fieldno,8))
227  laenge = VAL (fields$ (Fieldno,5))
228  par2   = aspalt
229 IF LEN (text$) > laenge THEN text$ = LEFT$ (text$,laenge)
230 IF LEN (text$) <= (par3 - par2) THEN
231  LOCATE azeile,aspalt
232  PRINT text$
233 ELSE
234  FOR tpos = 1 TO LEN (text$)
235    b$ = MID$ (text$,tpos,1)
236    LOCATE azeile,aspalt
237    PRINT b$
238    aspalt = aspalt + 1
239    IF aspalt > par3 THEN
240      aspalt = par2
241      azeile = azeile + 1
242    END IF
243  NEXT tpos
244 END IF
245 COLOR 1,0
246 END SUB
247
248 SUB DrawCursor (zeile,spalte,farbe) STATIC
249 'Zeichnet den Cursor in die angegebene Zeile und Spalte.
250 'FARBE = 1 : zeichnen, FARBE = 0 : löschen
251  x1 = (spalte- 1) * 8
252  y1 = (zeile - 1) * 8
253  LINE (x1+3,y1+3)-(x1+5,y1+5),farbe,b
254 END SUB
255
256 SUB GetField (Fieldno,text$) STATIC
257 'Eingaberoutine für das Feld Fieldno. Der eingegebene Text wird in text
    $ zurückgegeben. Vom Benutzer aufzurufen.
258  zei$   = fields$ (Fieldno,4)
259  prompt$= fields$ (Fieldno,12)
260  laenge = VAL (fields$ (Fieldno,5))
261  pzeile = VAL (fields$ (Fieldno,13))
262  pspalt = VAL (fields$ (Fieldno,14))
263  azeile = VAL (fields$ (Fieldno,0))
264  aspalt = VAL (fields$ (Fieldno,7))
265  par7   = VAL (fields$ (Fieldno,1))
266  par6   = VAL (fields$ (Fieldno,11))
267  par5   = VAL (fields$ (Fieldno,10))
268  par4   = VAL (fields$ (Fieldno,9))
269  par3   = VAL (fields$ (Fieldno,8))
270  par2   = aspalt
271  par1   = VAL (fields$ (Fieldno,6))
272  farbe  = 1
273 'Prompt ausgeben
274  IF pzeile > 0 THEN LOCATE pzeile,pspalt : PRINT prompt$
275  CALL MarkEOF (Fieldno,farbe)
276  IF (par1 AND 8) > 0 THEN COLOR 0,1:farbe = 0
277  IF farbe = 1 THEN bfarbe = 0 ELSE bfarbe = 1
278  text$ = ""
279
280  WHILE LEN (text$) < laenge
281    getit:
282    t$ = ""
283    CALL DrawCursor (azeile,aspalt,farbe)
284    LOCATE azeile,aspalt
285      WHILE INSTR(zei$,t$) = 0  OR t$ = ""
286        t$ = INKEY$
287        IF par5 = 1 THEN t$ = UCASE$ (t$)
288      WEND
289      IF t$ <> CHR$ (8) THEN GOTO weiter
290      ' Delete
291        IF LEN(text$) = 0 THEN BEEP:GOTO getit
292        text$ = LEFT$ (text$,LEN(text$)-1)
293        CALL DrawCursor (azeile,aspalt,bfarbe)
294        IF POS (0) = 1 THEN azeile = azeile - 1:aspalt = 77
295        IF POS (0) <> 1 THEN aspalt = aspalt - 1
296        LOCATE azeile,aspalt
297        PRINT " ";
298        LOCATE azeile,aspalt
299        CALL DrawCursor (azeile,aspalt,bfarbe)
300        GOTO getit
301      weiter:
302      IF t$ = CHR$(13) AND par4 = 2 THEN GOTO getit
303      text$ = text$ + t$
304      IF t$ <>CHR$ (13) THEN PRINT t$;
305      aspalt = aspalt + 1:IF aspalt > par3 THEN aspalt = par2:azeile = a
         zeile +1
306      IF t$ = CHR$ (13) THEN text$ = LEFT$ (text$,LEN(text$)-1):laenge =
          LEN(text$):PRINT " ";
307    WEND
308    COLOR 1,0
309    IF par6 > 0 THEN
310      CALL MarkEOF        (Fieldno,bfarbe)
311      CALL ClearField     (Fieldno)
312      CALL WriteIntoField (Fieldno,text$,0)
313    END IF
314    IF pzeile > 0 AND par7 = 1 THEN
315      LOCATE pzeile,pspalt
316      PRINT SPACE$ (LEN (prompt$) )
317    END IF
318 END SUB

(c) M&T
```

Listing 1. Die Unterroutinen von »Mask« (Schluß)

```
Programm : b:MaskDemo
---------------------

 1
 2 '----------------- DEMO -----------------
 3
 4 ' Felddefinitionen
 5 CALL SetField (0,"A-Z, ,-", 20, 5,10,29,139,"Vorname eingeben",4,10)
 6 CALL SetField (1,"A-Z, "  ,  1, 5,31,32,138,"Mittelname (Abkuerzung)",4
   ,31)
 7 CALL SetField (2,"A-Z, ,-", 30, 5,33,62,139,"Nachname eingeben",4,33)
 8 CALL SetField (3,"A-Z,0-9,., ",40, 6,10,49,139,"Straße und Hausnummer",
   7,10)
 9 CALL SetField (4,"0-9"      , 4, 7,10,13,138,"Postleitzahl (4 Ziffern)"
   ,8,10)
10 CALL SetField (5,"A-Z,0-9,-",48, 7,15,62,139,"Ort und Ortsteil",8,15)
11 CALL SetField (6,"A-Z,a-z,0-9,ö,ä,,Ö,Ä,Ü,-,.",400,7,10,70,141,"",0,0)
12 CALL SetField (7,"0-9,:",      5,10,38,42,116,"",0,0)
13 CALL SetField (8,"Y,J,N",      1,15,40,41,138,"Gefällt Ihnen das Program
   m ?",14,30)
14
15 'Eingaben
16 CLS
17 PRINT "        Demonstration des Unterprogrammpaketes MASK.UTILITY "
18 PRINT "        ---------------------------------------------------"
19 PRINT "1. Beispielanwendung : Addressdatei"
20 PRINT
21 PRINT "Name    :"
22 CALL GetField  (0,demo$ (0))
23 CALL GetField  (1,demo$ (1))
24 CALL GetField  (2,demo$ (2))
25 LOCATE 6,1
26 PRINT "Strasse:"
27 CALL GetField  (3,demo$ (3))
28 LOCATE 7,1
29 PRINT "Ort     :"
30 CALL GetField  (4,demo$ (4))
31 CALL GetField  (5,demo$ (5))
32 CLS
33 PRINT "        Demonstration des Unterprogrammpaketes MASK.UTILITY "
34 PRINT "        ---------------------------------------------------"
35 PRINT "2. Beispiel : Verschiedenes"
36 PRINT
37 PRINT "Bitte geben Sie einen beliebigen Text ein :"
38 CALL GetField (6,demo$ (6))
39 CALL GetField (8,demo$ (8))
40 CLS
41 PRINT "        Demonstration des Unterprogrammpaketes MASK.UTILITY "
42 PRINT "        ---------------------------------------------------"
43 PRINT "3. Beispiel : Verschiedenes"
44 LOCATE 9,1
45 PRINT "             Bitte geben Sie die Zeit (HH:MM) ein :"
46 CALL GetField (7,demo$ (7))
47 CALL GetField (8,demo$ (8))
48 IF demo$ (8) = "N" THEN
49 PRINT "Das gibt's ja wohl nicht !"
50 PRINT "Sie haben wohl nicht richtig hingeschaut !"
51 PRINT
52 PRINT "Aber Sie sollen noch eine Chance bekommen :"
53 ELSE
54 PRINT "Jawohl, Sie haben einen guten Blick, mit dem"
55 PRINT "Sie gleich die Qualität dieser Software richtig"
56 PRINT "beurteilt haben !"
57 PRINT
58 PRINT "Und weil's so schön war ..."
59 END IF
60
61 BEEP
62 FOR a = 1 TO 8000:NEXT
63
64 RUN

(c) M&T
```

Listing 2. Das Demoprogramm zu »Mask«

AMIGA
COMPUTER-MARKT

Wollen Sie einen gebrauchten Computer verkaufen oder erwerben? Suchen Sie Zubehör? Haben Sie Software anzubieten oder suchen Sie Programme oder Verbindungen? Der COMPUTER-MARKT von »Amiga« bietet allen Computerfans die Gelegenheit, für nur 5,— DM eine private Kleinanzeige mit bis zu 4 Zeilen Text in der Rubrik Ihrer Wahl aufzugeben. Und so kommt Ihre private Kleinanzeige in den COMPUTER-MARKT der **Oktober-Ausgabe** (erscheint am 30. September 87): Schicken Sie Ihren Anzeigentext bis zum 26. August 87 (Eingangsdatum beim Verlag) an »Amiga«. Später eingehende Aufträge werden in der **November-Ausgabe** (erscheint am 28. Oktober 87) veröffentlicht.

Am besten verwenden Sie dazu die vorbereitete Auftragskarte am Anfang des Heftes. Bitte beachten Sie: Ihr Anzeigentext darf maximal 4 Zeilen mit je 40 Buchstaben betragen. Überweisen Sie den Anzeigenpreis von DM 5,— auf das Postscheckkonto Nr. 14199-803 beim Postscheckamt mit dem Vermerk »Markt & Technik, Amiga« oder schicken Sie uns DM 5,— als Scheck oder in Bargeld. Der Verlag behält sich die Veröffentlichung längerer Texte vor. Kleinanzeigen, die entsprechend gekennzeichnet sind, oder deren Text auf eine gewerbliche Tätigkeit schließen läßt, werden in der Rubrik »Gewerbliche Kleinanzeigen« zum Preis von DM 12,— je Zeile Text veröffentlicht.

Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen

AMIGA

Suche Astrologieprogramm oder Informationen zum Selbstprogrammieren (Häuser nach Placidus), wenn möglich Münchner Rhythmenlehre (auch IBM-Programme). Tel. 0721/698944

Suche Tauschpartner auf dem Amiga und C64!!! Schickt eine Liste oder 1-2 Disks mit den neuesten Prog. Sandman, Danzigerstr. 26, 7406 Mössingen 3, TWW 1993

AmigaPublic-Domain-Software, 10 Disk 65 DM, 20 Disk 120 DM, über 30 Disk Stück 4.50 DM. Alles mit Disk und Anl. Tel. 02056/60562, ab 19 Uhr, Frank

Verkaufe MS-DOS-Transformer & Systemdiskette (Original!) für 90 Mark. *Suche C-Compiler*, E. Galler, Bretzenheimer Str. 26, 6500 Mainz

Suche Tauschpartner für Software und 3,5-Zoll-Disk (135 tpi), wenn möglich doppelseitig. Bitte melden unter: Tel. 05103/7507

CMP/Hallo Freaks. Wir vermitteln fast alles, was Computer betr. Suchen: Musikpro.-, Disks, Hard-Software, 50 PF an: CMP, Wiechertstr. 34, 4030 Ratingen 1, alles klar./CMP

Verk. 768 KB RAM-Erweiterung incl. Kickstart (modifiziert) vorne steckbar f. 440 DM*** TV-Kontrastfilter f. 1081 (augenschonend u. blendfrei) für 110 DM. Tel. 089/963250 ab 19 Uhr.

Verk. PAL-Amiga 1000 m. 1 MB RAM, Monitor* 1081, 10 Progr. Disks u. Handbuch f. 2200 DM, Bildschirmkontrastfilter f. 1081 (augenschonend u. blendfrei) f. 110 DM, Tel. 089/963250 ab 19 Uhr.

Amiga 1000 + 3½-Zoll Floppy von NEC + Star Drucker SG 10 + 100 Disketten + Bücher + Zubehör. VB 3950 DM, Oberhausen, Tel. 0208/808461

DACG-Der Computerclub!!! Wir bieten: Informationen, Software, Kontakte, Einkaufsvorteile, Clubzeitschrift! Kein Mitgliedsbeitrag** DACG, LA-Bazoge 342, 2811 Martfeld

Verkaufe Amiga 1000 mit Sidecare, 2. Laufwerk, Tintenstrahldrucker und div. Software, VB 4900 DM, Michael Haerle, Schornreuteweg 56, 7980 Ravensburg, Tel. 0751/25980

Verkaufe int. Floppy für Amiga-2000 wegen Doppelbestellung 310 DM (neu). P. Engels, Kreisstr. 29, 5308 Rheinbach 14

Amiga 500/1000/2000 Software vorhanden. Tauschpartner gesucht, auch Anfänger. 256 K-Erweiterung zu verkaufen, 80 DM. Postfach 1129 in 8901 Königsbrunn.

Suche billig!! Anwender-Software für Commodore Amiga 2000, Tel. 0251/231835, Mo.-Fr. 13.30 Uhr bis 14.45, 19 bis 22 Uhr oder schriftlich, Th. Geisbe, Kärntnerstr. 45, 4400 Münster

Suche alles an Public Domain für Amiga. Ankauf oder Tausch, habe über 150 Disks. Gründe eine P-D-Bibliothek. M. Rönn, Ziegeleiweg 32, 3257 Springe, Tel. 05041/8229

Suche Kontakt zu netten Amiga-Anwendern, auch Neulingen, zwecks Erfahrungs- und Programmtausch. Ic h nutze mein Gerät privat und geschäftlich. Michael Lange, Bappstr. 4, 1000 Berlin 61, Tel. 030/6931028

Verkaufe für Amiga 1000 einen Aufsatz für 2. bis 4. Laufwerk. Aussehen wie Amiga 2000 für 50 DM, Info + Bild für 1 Leerdisk. Hasenbeck, Vereinstr. 23, 5780 Bestwig

A500-Neuling sucht zuverlässigen Softwaretauschpartner. Angebote an: Holger Kaczmarek, Turnhallestr. 64, 7290 Freudenstadt

Wer verkauft mir seine Originale zu günstigem Preis? Spiele bevorzugt. Suche u.a. The Pawn, SDI, Atlantis, Mindshadow. Keine Raubkopien!!!, Tel. 05204/8196

Sidecar 512 K für Amiga mit V20-Prozessor und Disketten für 1400 DM bei Jürgen Brockmann, Brunnenstr. 14, 5591 Ernst, Tel. 02671/1273 ab 18 Uhr.

Amiga 500: Suche Tauschpartner für Amiga Programme. Suche MS-DOS, PD-Disks. Roland Neher, Talstr. 11, 7960 Aulendorf, Tel. 07525/8913

Tausche immer neueste Amiga-Software, Tel. 02821/6350

Suche Sidecar für Amiga 1000, habe + suche beste Hard+Software. Koantakte im In- und Ausland erwünscht. Bis bald!!! Charly, Tel. 04421/34754

Amiga Suche und tausche Public Domain-Software. Habe Fish 1-58 etc. Kein Gewinninteresse, sehr billig. Tel. 06139/6374

Verkaufe Zeitschriften und Sonderhefte: 68000'er, 64'er, Happy, Data Welt, ASM... Liste gegen 80 PF, Rückporto. Martin Brill, Jakobstr. 36, 6688 Illingen

Verkaufe kompl. Amiga 1000-Station!!! Mon. 1081 + Drehständer + Filter/orig. Disks m. M&T-Handbuch/Pal-Version, k. deut. Tastatur, VB 2000 DM, Tel. 089/393595, 14 bis 17 Uhr

Suche Kontakte zu Amiga 1000-Besitzern im Raum München 21 (Laim) oder Umgebung zum Erfahrungsaustausch! Bitte melden: Tel. 089/5704331 ab 18.30 Uhr.

Modula 2, wer beschäftigt sich auch mit dieser Sprache auf dem Amiga?? Bitte melden bei F. Schäfer, Tel. 06035/4439, Lessingstr. 9a, 6361 Reichelsheim

Verkaufe für den Amiga: Amiga-DOS Manual von Bantam-Books (engl), Preis VB. Suche Gunship und Jet (A 500). Tel. 07564/3479

Amiga-Software? Call: 02173/71476

AmigaAmiga***Amiga***
Top-Programme (teilweise mit Anleitung) verkauft preiswert: Dietmer, Rainer, Am Sonnigen Hang 4, 6500 Mainz 32

Verk. Amiga 1000, Drucker Epson FX 85, Kontakt zu Amiga-User gesucht (auch Raum Rosenheim), Tel. 08031/15507 ab 18 Uhr.

Ich suche Kontakt zu anderen Amiga-Progammierern. (Spez. im Raum HD-MA) Tel. 06227-4699 Bernhard

Suche Tauschpartner für Software (auch Anfänger). Tel. 040/6516762

Achtung!! Suche zuverlässige Tauschpartner für Amiga-Software! Ca. 80 Disketten vorhanden. Nur ernstgemeinte Zuschriften. An alle armen Schüler! Geht lieber lernen. Frank Müller, Bottrop, Tel. 02041/687260

Habe/Suche Amiga-Software und Anleitungen (neueste). Bitte melden bei: Jean-Jacques Wagner, Postfach 7, 3807 Iseltwald (Schweiz) ***Neueste***Achtung! Bitte keine Anrufe....

Suche Original-Software, MIDI-Interface, Centr. Kabel f. A1000. Liste m. Tel. Nr. u. Preisvorstell. an Rainer Dunkelgut, Neuenteilsredder 15, 2400 Lübeck 14

Amiga-JUICE. Das Amiga Magazin auf Diskette!!! Amiga-JUICE ist Public-Domain und jeder kann mitmachen! Ruft einfach an: 05103/2146

Suche Grafiker mit Amiga für Erfahrungsaustausch u. Problemlösung. Raum F., HG. Demo-Disk m. Business-Grafiken, einige Programme doppelt, Kontakt 06081/8590 ab 19 Uhr.

Suche Tauschpartner für Software; suche ebenfalls billige und gute Erweiterungen jeder Art. Hauptsächlich Speichererweiterungen. Tel. 05103/7507

AmigaAtari***C64***Apple***IBM*** Wir suchen noch Tauschpartner! Haben die neuesten Games-Intros aus USA (k. Anfänger). A. Höhn, Am Schwalbenhang 3, 7730 Villingen

Public-Domain-Software zum Aufbau einer PD-Bibliothek gesucht. (Tausch u. Ankauf) Klaus G. Seidler, Tölzer Str. 14, PF 561, 1000 Berlin 33, Tel. 030/8231793

Hallo Amiga-User, suche Tauschpartner für Software. Immer aktuelle Software vorhanden. Tel. 02904/2847 ab 17 Uhr

****Achtung****
Amiga 1000 + Monitor 1081 + EXT.Laufw. + LIT. + Software. Kaum gebraucht! Preis VB ab 17 Uhr. Tel. 089/855899 oder 089/853844

Amiga**Public-Domain-Software** Tausche PD-Software, Peter Schomann, Südstr. 16, 4784 Rüthen 5, Tel. 02954/1050

Suche für Amiga 500 Orig.-Programme mit DT.Anleitung. Suche auch DT.Anleitung für Textcraft+Graficraft. Liste mit Preise an: Widmayer, Neckarstr. 51, 7148 Remseck 2

Suche Kontakt zu Sidecar-Besitzern zwecks Informations-Austausch. Briefe an: Nicol Mayer, Johannes-Biegstr. 2, 7121 Erligheim

Münster — Osnabrück. Suche Kontakt zu Amiga-Anwendern und deutsche Anleitungen. H. Wolf, Tel. 05426-557

Suche Public Domain-Soft deutschsprachiger Autoren für deutschen PD-Pool-Programme mit Sourcecode bitte senden an: Uwe Trempelmann, Lochnerstr. 24, 4030 Ratingen

PDS for Amiga! Laßt die Softwarewelle rollen! Tausche und suche ständig Public-Domain-Software. Lutz Beyert, Eckampstr.4, 4030 Ratingen

Achtung! Suche billige Software für Amiga! Angebote an: Sven Kronberg, Habichtstr. 2, 6835 Brühl, Go Amiga!!

Wer kauft mit mir Orignalprogramme? (Schwerpunkt: Anwender-PRGs., aber auch sehr gute Spiele) Meldet Euch! Rainer Lienhart, A.-Schmitt-Str. 20, 6400 Fulda, Tel. 0661/54772

Pro-Writer-Textverarbeitung, Originaldiskette und Handbuch, neueste Version für 185 DM. Tel. 02158/6856. Nur Sa. u. So.

RAM**RAM**RAM**RAM**RAM
256 KB-Fronterweiterung Amiga billigst abzugeben: 150 DM (VB). Tel. 0671/40110, Amiga ist der Untergang des Atari ST!!!

Amiga*Schüler sucht preisgünstigen Amiga. Wenn möglich geschenkt. Bitte melden bei: D. Lederer, Zeppelinstr. 9, 8600 Bamberg, Wanted: Amiga 500, 1000 oder 2000 **Amiga**

Amiga Neuling mit Basic und Assemblerkenntnissen sucht Amiga Club in Dortmund, Ingo Kühler, Aplerbecker-Mark-Str. 16a, 4600 Dortmund 41, Tel. 0231/483382

Verschiedene gebrauchte Computer z. verk., CBM-8032 + Floppy-8050 + Adcomp-X80SP, 1000 DM, BASF7100 + 2 Floppys + Programme für 500 DM. ORIC-ATMOS-1+2+10 Prog. 350 DM, Tel. 0791/52481

Tausche Amiga-Software — So call quick: Tel. 0221/896109 ab 15 Uhr, Frank

Ausland

Suche Horoskop- und Astrologieprogramme zu kaufen. Auch Tauschpartner aus allen Ländern. Zuschr.: F. Franzwa, A-5015 Salzburg, Postfach 24

Suche Tauschpartner: Ich habe viel Amiga-software zum Tauschen, also meldet Euch: Markus Vetterli, Schulstr. 22a, CH-5415 Nussbaumen/Schweiz

Suche Tauschpartner für Amiga Software und Manuals. Listen an: Peter van Dam, De Heygraeff 11, 3931 MK Woudenberg, Holland

SSD of TSK-Crew is now also on Amiga! Contact me for Swapping Software. Write to: Martin Gössnitzer, Unteragsdorf 25, A-9433 St. Andrä. Tel. 0043/43583047

Amiga-Freaks, meldet Euch!! Suche Tauschpartner für Amiga-Software und Anleitungen. Schreibt an Gerhard Zweimüller, A-4906 Eberschwang 113

Austria-Only-Austria-Only-Austria-Only, Amiga-Partner gesucht! Robert Graf, Bahnhofstr. 34, A-3462 Absdorf, Te. 02278/26815

2 MB-Speichererw. m/recoverable RAM-Disk SFr. 1200. Kompl. Satz 4 Stk. Reference Manuals (Amiga-H. 6/7 S. 125) SFr. 195. VIP-Prof. orig. 295 DM. Tel. 0041/22765601

Schweiz, Basel, Schweiz. Verkaufe 6 Monate alten EPSON FX85-Drucker, NP 1450 DM jetzt nur noch 890 sfr. Melden bei Calligaris, Tel. 061/498827 oder im Dial-Club. Tel. 061/225167

GEWERBLICHE KLEINANZEIGEN

Public-Domain-Software, IBM, Atari, Amiga über 3000 ausgesuchte Prog.Disks ab 5 DM. Liste frei. HCT Computertechnik Borbeckerstr. 127, 4300 Essen, Tel. 0201/687830

Fred Fish Public Domain Software Library. 31 Fish-Disks+Ausf. Dtsch. Anleitung + **Garantie** auf HQ-Disks, zum fairen Preis von 165 DM! V-Schk: B. Schilder, Laarkamp 6, 4630 Bochum 6

Futurevision: Hard- und Software (Laufwerke, Megas, Disks, Trackscanner, News aus USA, Public Domain). Herstellung und Vertrieb. Ab 18 Uhr. Tel. 06163/4326!!!

Super Amiga Public Domain-v. AIT. Alles von: Fish, Faug, Panorama ect. Info-Disk mit Beschr. aller Prog. (mit über 250 Bildschirmseiten) 10 DM Vorkasse. Wird mit 1 bzw. 2 Extra-Disks verrechnet/Berechtigt zu Sonderpreisen bis zu 5 DM pro Disk. M. Rönn, Ziegeleiweg 32, 3257 Springe 4, 05041/8229/19 Uhr.

Teac-Laufwerke für Amiga ab 320 DM, Disketten ab 1 DM., Tel. 07121/61882

Amiga 500/1000/2000 Public Domain: Ausführlicher Gratiskatalog bei: PD-Soft, Pf. 359, 4290 Bocholt

Amiga Echtzeituhr 149 DM, für Sidecar 89 DM. Tel. 030/6024249

Public-Domain — mehr als 250 Disketten, Preise je nach Abnahmemenge bis 4,80 DM. Katalogdiskette gegen 5 DM anfordern. Stefan Ossowski, Veronikastr. 33, 4300 Essen

Tips & Tricks zum Amiga

Unser Aufruf, sich an der Gestaltung dieser Rubrik zu beteiligen, hat breite Zustimmung gefunden. Wir haben eine Auswahl — es sind Tips zum Umgang mit CLI, Drucker und Batchfiles sowie kleine Assembler-Routinen dabei — für Sie zusammengestellt.

Zu den bereits veröffentlichten Tips & Tricks der letzten Ausgabe hat uns so mancher Verbesserungsvorschlag erreicht. Das ist auch gut so. Diese Seite lebt von Ihnen. Sie soll ein Forum für den Gedankenaustausch der Amiga-Fans sein. Zögern Sie nicht, uns zu schreiben, wenn Sie der Meinung sind, daß Ihre Erfahrungen nützlich sind für die Freunde Ihrer Freundin (Amiga). Jeder brauchbare Beitrag wird in dieser Rubrik veröffentlicht.

Maus stoppt Bildschirmausgabe

Durch Druck einer beliebigen Zeichentaste lassen sich Bildschirmausgaben anhalten und mit <Backspace> wieder fortführen. Den gleichen Effekt hat das Betätigen der Menütaste (rechts) der Maus. Der Bildschirm und damit die Ausgabe wird eingefroren und kann in Ruhe betrachtet werden.

(Hesselbart/pa)

Druckeransteuerung

Wenn man versucht, seinem Drucker über die Druckerschnittstelle mehr als nur Text, also auch Steuerzeichen zu übermitteln, so kann man schnell verzweifeln... denn da kommt nichts an!

Der Grund liegt in der speziellen Druckersteuerung des Amiga. Es wurde nämlich — um die 100prozentige Kompatibilität aller Programme zu gewährleisten — ein spezieller Druckercode entwickelt, der dann von dem mit Preferences gewählten Druckertreiber für den jeweiligen Drucker umgesetzt wird. Das ist eine sehr intelligente Idee, denn so erspart man sich die umständliche Installation von neuen Programmen.

Im Handbuch wird nur kurz — durchaus üblich für Commdore-Dokumentation — auf diesen Umstand eingegangen. Hier nun eine Auswahl der meistverwendeten Drucker-Steuer-Codes (ESC steht für CHR$(27); die übrigen Zeichen sind in Anführungsstriche zu setzen; es können auch mehrere Befehle kombiniert werden, diese sind dann durch einen Strichpunkt zu trennen):

ESC[2w - Elite ein
ESC[1w - Elite aus
ESC[4w - Schmalschrift ein
ESC[3w - Schmalschrift aus
ESC[6w - Breitschrift ein
ESC[5w - Breitschrift aus
ESC[2"z - NLQ ein
ESC[1"z - NLQ aus
ESC[2p - Proportionalschrift ein
ESC[1p - Proportionalschrift aus
ESC[2v - Superscript ein
ESC[1v - Superscript aus
ESC[4v - Subscript ein
ESC[3v - Subscript aus
ESC[1m - Fettdruck ein
ESC[22m - Fettdruck aus
ESC[4m - Unterstreichen ein
ESC[24m - Unterstreichen aus

Ein Beispiel für eine Textausgabe in Fettdruck:

```
LPRINT CHR$(27);"[1m";"Hallo Welt!"
```

Ein weiteres für die Übergabe des Anführungszeichens:

```
LPRINT CHR$(27);"[";chr$(34);"z";"Diesmal in NLQ"
```

Wer diese Art der Druckersteuerung in seinen Programmen verwendet, gewährleistet dadurch, daß sein Programm auch auf anderen Druckern die gleichen Ergebnisse liefert. Wem diese Art der Programmierung jedoch zu umständlich ist, der kann den Drucker ohne die Umleitung über »prt:« auch direkt ansprechen, indem er die Schnittstellen »par:« oder »ser:« verwendet. Alle Steuerzeichen werden dann »unzensiert« übermittelt. Im »ROM Kernel Manual: Libraries and Devices« befindet sich im Kapitel 15 (Printer Device) eine komplette Liste dieser Drucker-Kontrollfunktionen. Leser mit C-Kenntnissen erhalten hier auch detaillierte Informationen über den Umgang des Amiga mit einem Drucker.

(Michael Baas/pa)

Setzen der Echtzeituhr

Der Amiga 2000 besitzt neben der von den anderen Amigas bekannten Systemuhr eine akkugepufferte Echtzeituhr. Diese Uhr läßt sich jedoch weder über »Preferences« noch über »Date« einstellen. Auf der Workbench befindet sich für diesen Zweck das Programm »Setclock«. Das mitgelieferte Handbuch schweigt sich über seine Bedienungsweise allerdings aus. Hier die Erklärung: »setclock opt load« lädt die Uhrzeit der akkugepufferten Uhr in die Systemuhr des Amigas. »setclock opt save« speichert die Uhrzeit des Amigas in die akkugepufferte Uhr ab. Beide Befehle übertragen natürlich auch das Datum. Vor dem Laden der gepufferten Uhr ist die Systemuhr wie gewohnt mit »Date« zu laden.

(Stefan Berner/pa)

Schneller Druckertreiberwechsel

Dieser Tip richtet sich an diejenigen, die zwei oder mehr Drucker haben und denen der Druckertreibertausch mit »Preferences« zu umständlich ist. Es wurde für Kickstart 1.2 entwickelt und sollte außerdem nicht mit der Original-Workbench nachvollzogen werden. Um die Schritte zu verdeutlichen, verwenden wir im Beispiel die Treiber für den Epson JX-80 und den Diablo C-150.

1. Starten Sie den Amiga wie gewohnt mit Kickstart und Workbench. Sobald der Text »Workbench-Diskette. Version ...« erscheint, drücken Sie bitte <Ctrl-D>. Sie befinden sich nun im Fenster des CLI (Command Line Interface).
2. Geben Sie folgende Befehle ein und lösen Sie diese, wie alle anderen, mit <Return> aus: »system/SETMAP d« und »LOADWB«. Sie befinden sich nun auf der Workbench.
3. Lassen Sie sich mit »DIR devs/printers« die auf der Diskette befindlichen Treiber ausgeben und merken sich den Namen derjenigen, die Sie öfter verwenden wollen.
4. Kopieren Sie die Treiber in das Hauptdirectory. Beispiel: »COPY devs/printers/Epson_JX-80 TO Epson_JX-80«
»COPY devs/printers/Diablo_C-150 TO Diablo_C-150«
5. Nun schaffen Sie erst mal Platz auf der Diskette. Das können Sie mit der Anweisung »DIR devs/printers OPT I«. Es erscheinen die einzelnen Namen der Treiber und Sie können durch Eingabe des Wortes »del« hinter dem Fragezeichen alle Treiber, die Sie nicht mehr brauchen (normalerweise alle), löschen.
6. Leihen Sie sich mit »COPY clock.info to Epson_JX-80.info« das Icon der Uhr für den ersten Treiber. Führen Sie diese Anweisung für jeden der vorher kopierten Treiber durch.
7. Mit der Anweisung »ENDCLI« gelangen Sie nun wieder auf den Bildschirm der Workbench. Öffnen Sie die Workbench-Disk, wählen dort die Empty-Schublade und im Menü den Punkt Duplicate aus. Im Fenster erscheint eine Schublade namens »Copy of Emp-

ty«. Geben Sie ihr mit »Rename« aus dem Menü den Namen »Drucker«. Nach dem Öffnen dieser Schublade können Sie das Uhren-Icon mit den Treibernamen in die neue Schublade ziehen. Wenn Sie jetzt mal wieder den Treiber wechseln wollen, so reicht ein Anklicken in der Workbench, um ihn auszuwählen.
Wenn Sie sich mit dem Icon-Editor auskennen, können Sie auch das Icon ändern. (Peter Riedlberger/pa)

Keine Probleme mit »Disk Full«

Wenn beim Abspeichern von Daten oder Programmen diese Systemmeldung erscheint, werden Sie nicht nervös, sondern ignorieren erst einmal diese Meldung. Sie haben ja ein Multiprogrammsystem, bei dem es keine Rolle spielt, wenn ein Programm nicht weiterarbeiten kann. Eröffnen Sie irgendwo ein neues CLI-Fenster durch Anklicken des CLI-Icons und löschen nicht mehr benötigte Files auf der vollen Diskette. Sie können zwischenzeitlich auch die Workbench-Diskette einlegen. Der Amiga vergißt Ihre volle Diskette schon nicht. Selbstverständlich können Sie auch Files auf eine andere Diskette kopieren und danach erst löschen. Wenn Sie der Auffassung sind, daß jetzt genügend Platz auf der Diskette ist, klicken Sie in der Systemanforderung mit dem frechen »Disk Full« das Retry-Feld an; der Computer speichert weiter. Vielleicht kommt die Meldung erneut; wiederholen Sie einfach das beschriebene Verfahren. (Paul Scholda/pa)

Bildschirmscrollen in »C«

Mit dem Befehl »ScrollRaster(rp,x,y,xmin,ymin,xmax,ymax)« läßt sich in »C« der Bildschirm auf einfache Weise scrollen. Dabei ist »rp« ein Pointer auf ein Rasterport, x und y sind die Anzahl der Punkte, die in x- oder y-Richtung gescrollt werden sollen und die restlichen vier Parameter die Koordinaten des zu scrollenden Bildschirmausschnittes. (Cord-Henning Scholz/pa)

DOS-Befehle umbenennen?

Es ist nicht unbedingt ratsam, seine DOS-Befehle einfach umzubenennen. Erstens gibt es elegantere Lösungen und zweitens schafft man sich früher oder später Probleme damit: Was passiert, wenn andere Programme auf die CLI-Befehle zugreifen wollen? Bei intensiver Verwendung von Abkürzungen ergibt sich bald die Frage: War »FA« jetzt »FAULT« oder »FAILAT« (oder ähnlich) ...? Es ist zugegebenermaßen umständlich, für das Starten eines Batch-Files erst »EXECUTE« eingeben zu müssen, deshalb kann man in der Startup-Sequence die Anweisung »ASSIGN x: c:EXECUTE ergänzen. Sie sorgt dafür, daß man, um den Befehl EXECUTE aufzurufen, künftig nur noch x: eingeben muß — unabhängig davon, ob sich die DOS-Befehle auf der RAM-Disk oder sonstwo befinden.

Wer es jedoch partout noch kürzer will, der sollte sich auf der Workbench ein Directory mit dem Namen »My_Commands« anlegen, den Befehl EXECUTE dort hineinkopieren und mit RENAME abkürzen. Nach der Eingabe von »PATH ADD SYS:My_Commands« findet das CLI auch die abgekürzten Befehle wieder. Die Erklärung der Befehle: Mit dem ASSIGN-Befehl kann man Dateien und Geräten quasi Synonyme zuweisen, mit denen die Dateien auch aufgerufen werden können. Der Doppelpunkt hinter dem »X:« ist dabei erforderlich. Einen Überblick über die aktuellen Zuweisungen erlangt man durch ASSIGN ohne Parameter. Das Löschen der Zuweisung ist mit der Anweisung ASSIGN unter Angabe des Geräte- beziehungsweise Filenamens möglich (beispielsweise ASSIGN x:) Mit der Anweisung »PATH ADD« lassen sich zusätzliche Directories angeben, in denen das CLI nach seinen Befehlen sucht. Standardmäßig sind dies nur das Directory c: und das jeweils aktuelle Directory. (Michael Baas/pa)

Programmabbruch in Assembler

Bei manchen Assembler-Programmen ist nach Ausführung eine sofortige Rückkehr in das System oder, wenn das Programm vom Assembler aus gestartet worden ist, in den Assembler nicht wünschenswert. Vielmehr möchte man sich vielleicht am Bildschirm die Resultate der Programmausführung, eine Grafik oder ähnliches, erst in Ruhe anschauen. Irgendwann sollte das Programm allerdings zurückkehren. Die folgende Routine zeigt ein Beispiel mit einer möglichen Lösung dieses Problems. Aus der Adresse $dff00a wird die Position des Mauszeigers auf dem Bildschirm gelesen und in das Bildschirmfarbregister geschrieben.

```
LOOP: MOVE $DF00A,$DFF180
      ANDI.B #64,$BFE001
      BNE LOOP
      RTS
```

Durch Bewegen der Maus kommt nun auf dem Bildschirm Farbe ins Spiel. Der eigentliche Kern — das Testen der Speicherstelle, in der ein Druck auf die linke Maustaste registriert wird — mit dem ANDI-Befehl beendet das Programm nach einer entsprechenden Betätigung der Taste. (Martin Dencker/pa)

Batchfiles beschleunigen

Unter einem Batchfile versteht man eine Textdatei, die CLI-Befehle enthält. Man kann in diese Datei eine Reihe solcher Befehle unterbringen und mit »EXECUTE Name des Batchfiles« werden diese Kommandos automatisch der Reihe nach ausgeführt. Die CLI-Befehle sind Programme wie jedes andere Anwenderprogramm auch — sie müssen also erst von der Diskette geladen werden (sofern sie sich nicht in der RAM-Disk befinden). Die Ladezeiten fallen besonders ärgerlich auf, wenn in einem Batchfile mehrere ECHO-Befehle (funktioniert deshalb ähnlich PRINT in Basic) nacheinander auftauchen und jedesmal neu geladen werden müssen. Ähnlich verhält es sich mit anderen Kommandos bei der täglichen Arbeit mit dem CLI (CD...DIR,CD...DIR und so weiter). Neben der Möglichkeit, die Befehle in die schnelle RAM-Disk zu legen, gibt es eine weitere, das RAM einzubeziehen. Man ordnet dem Laufwerk DF0: einen bestimmten RAM-Bereich als Puffer zu, in dem die zuletzt geladenen Befehle gespeichert werden. So brauchen diese Kommandos nicht neu geladen, sondern können aus dem Puffer gelesen werden. Um mit dem Befehl
ADDBUFFER Laufwerk Puffergröße
wirklichen Zeitgewinn erzielen zu können, sollte man den Puffer etwa 18-20 KByte groß wählen. Beispiel: ADDBUFFERS df0: 18. Der einzige Nachteil dieser Methode soll jedoch nicht verschwiegen werden. Es ist nur durch einen Reset möglich, diesen Speicherbereich wieder anderweitig zu vergeben — man sollte sich also genau überlegen, wieviel Speicher man dafür »opfert«.
(Michael Baas/pa)

Mit beiden Armen tief im System

Hier noch ein paar kleine Routinen für die Assembler-Freaks: Abfrage der rechten Maustaste:

```
LOOP: BTST #10,$DFF016
    BNE.S LOOP
```

Power-LED dunkel:

```
    OR.B #2,$BFE001
```

Power-LED hell:

```
    ANDI.B #253,$bfe001
```

Unterroutine für die Initialisierung einer eigenen Copperliste:

```
set_user_copperlist:
        MOVE.L 4,A6
        LEA Gfxname,A1
        JSR -408(A6)
        MOVE.L D0,A5
        ADD.L #$32,A5
        MOVE.L A5,savecop2
        MOVE.L (A5),savecop
        MOVE.L #COPLIST,(A5)
        RTS
```

Unterroutine für das Entfernen einer eigenen Copperliste:

```
recall_old_copperlist:
        MOVE.L savecop,A5
        MOVE.L savecop2(A5)
        RTS
```

Label, die von den Copper-Routinen gebraucht werden:

```
gfxname: dc.b "graphics.library",0
even
savecop: dc.l 0
savecop2: dc.l 0
```

(Warnung der Redaktion: Sie sollten wissen was Sie tun, wenn Sie in dieser oder anderer Form direkt auf die Hardware des Amiga zugreifen.)

(Christian Zaenker/pa)

Eine neue Startup-Sequence

Angeregt durch den Beitrag »Amiga als Schreibmaschine« der letzen Ausgabe hier eine etwas komfortablere Methode:

Mit der Änderung der Startup-Sequence einer Workbench kann man den Amiga gleich nach dem automatischen Starten (Booten) der Diskette als Schreibmaschine verwenden. Die Startup-Sequence ist ein Batchfile, das beim Booten automatisch ausgeführt wird. Folgende Schritte sind auszuführen:

1. CLI der Workbench anklicken und die folgenden Anweisungen eingeben: CD df0: und ED h
2. Sie befinden sich nun im Editor. Schreiben Sie dort die Zeilen:
ed b
copy b to prt:
delete b
Mit <ESC x> und <Return> wird die Datei gespeichert und Sie kehren zurück in das Fenster des CLI.
3. Nun wird der Editor erneut mit »ED s/startup-sequence« aufgerufen. Auf dem Bildschirm erscheint die Startup-Sequence. Fügen Sie hinter »LOADWB« die Befehle CD df0: und EXECUTE h ein. Wiederum wird der Editor mit <ESC x> verlassen.

Nach dem Booten der Workbench gelangen Sie jetzt sofort in den Editor, schreiben Ihren Text und verlassen ihn wie gewohnt mit <ESC x>. Der Text wird dann über die parallele Schnittstelle auf den Drucker ausgegeben. Sollte Ihr Drucker sich am seriellen Port des Amiga befinden, so schreiben Sie in der Startup-Sequence ser: anstatt prt:. Wenn Sie einen zweiten Brief schreiben möchten, so kann der gleiche Vorgang durch EXECUTE h erneut gestartet werden.

(Karl Heinz Theobald/pa)

CLI im Rausch der Farben

Vielleicht haben Sie sich schon einmal gewünscht, die verschiedenen Schrifttypen (Kursiv, Fett, Invers und Unterstrichen) in den Bildschirmausgaben der CLI-Befehle verwenden zu können? Das Einstellen der Vordergrund- und Hintergrundfarbe für Ausgaben des CLI wäre auch von Vorteil.

Commodore erwähnt in dem ohnehin nicht üppigen DOS-Manual mit keiner Silbe, daß so etwas möglich ist. Man muß lediglich in die Ausgabebefehle (ECHO, PROMPT, FILENOTE und andere) eine Escape-Sequence einbauen und schon kann man alle Schrifttypen in den verschiedensten Farben benutzen. Der Aufbau einer solchen Sequenz sieht so aus:

SEZ Schrift [; Vordergrund [; Hintergrund]]m

Dabei bedeutet SEZ die Sequenz-Einführungszeichen (*,e,]). Hinter den Einführungszeichen wird die Schriftart gewählt:

0 = normal
1 = Fettdruck
3 = Kursiv
4 = Unterstrichen
7 = Invers

Für die Vordergrundfarbe sind folgende Parameter erlaubt:

30 = normal
31 = weiß
32 = schwarz
33 = orange

Und schließlich für die Hintergrundfarbe:

40 = normal
41 = weiß
42 = schwarz
43 = orange

Als Endekennzeichen fungiert das Zeichen <m>. Auch die Kombination von verschiedenen Schriftarten ist durch mehrfache Angabe solcher Sequenzen möglich.

Für die Ausgabe eines unterstrichenen kursiven Textes in Orange auf weißem Grund ist also einzugeben:

ECHO "*e[3;4;33;41m Test-Text *e[0m"

Wie im Beispiel sollte man die Schriftart am Schluß wieder mit der Folge »*e[0m auf normal einstellen. Übrigens wirken diese Einstellungen auch auf den Drucker, wenn man die Ausgabe mit >prt: (Drucker am Parallelport) oder >ser: (serielle Schmittstelle) in derselben Anweisung umlenkt.

(Michael Baas/pa)

Vorteil durch weniger Speicher

Im Amiga 2000 befindet sich eine Speichererweiterung auf der sogenannten Uhrenkarte. Diese Karte kann im Amiga 500 nachgerüstet werden. Beide Computer haben dadurch 1-MByte-Speicher. Man bezeichnet diesen zusätzlichen Speicher auch als »Fast Memory«, da die Customchips (Agnus, Paula, Denise) diesen Bereich nicht adressieren können und deshalb den Prozessor beim Zugriff nicht behindern. Leider funktionieren nicht alle für den Amiga 1000 entwickelten Programme mit dieser Erweiterung. Für diesen Fall enthält die Workbench dieser Modelle 500 und 2000 das Programm »NoFastMem«. Es wird durch die Anweisung »run nofastmem« gestartet und schaltet den erweiterten Speicherbereich praktisch ab. Trotzdem funktionieren dann noch nicht alle Programme. Abhilfe schafft in einigen (nicht allen) Fällen ein erneutes Starten von NoFastMem.

(pa)

Please replace Workbench . . .

Vielleicht haben Sie sich über die gut gemeinte Aufforderung auch schon geärgert, wenn Sie ein auf einer anderen Diskette gespeichertes Programm starten, aber beim besten Willen nicht das auf der Workbench befindliche Betriebssystem und zugleich das gewünschte Programm im Laufwerk haben können, weil Sie kein externes Laufwerk besitzen. Ein zweites Laufwerk ist daher eine der wichtigsten Erweiterungen, die man seiner Freundin (sprich: Amiga) gönnen sollte. Das billigste, schnellste und leiseste Laufwerk auf dem Markt ist die RAM-Disk. Sie belegt allerdings kostbaren Speicherplatz. Wie wird man eine RAM-Disk, wenn man sie nicht mehr benötigt, wieder los?

Praktisch wäre das Einrichten einer solchen Disk und beliebiges Entfernen dieser vom CLI aus. Mit den beiden folgenden Programmen läßt sich dies auch realisieren:

Legen Sie sich mit der Anweisung »MAKEDIR df0:ram« auf der Diskette ein Verzeichnis mit dem Namen »ram« an. Aktivieren Sie dann den Editor mit »ED df0:ram/on« und tippen die folgenden Zeilen ein:

```
makedir ram:c
copy :c/copy ram:c/copy
assign c: ram:c
copy :c/dir ram:c/dir
copy :c/cd ram:c/cd
copy ...
```

Welche und wieviel Betriebssystem-Befehle in das Verzeichnis »ram:c« kopiert werden, liegt bei Ihnen; auf jeden Fall sollten sich die Befehle DIR und CD darin befinden.

Mit <ESC-x> verlassen Sie den Editor und haben soeben eine Batch-Datei erstellt. Mit der Anweisung EXECUTE ram/on werden nun die in der Datei befindlichen CLI-Befehle der Reihe nach ausgeführt. Mit dem Befehl INFO können Sie sich davon überzeugen, daß Ihnen nun zwei Laufwerke zur Verfügung stehen. Da der DIR-Befehl sich jetzt hoffentlich in der RAM-Disk befindet, können Sie sich ohne Probleme Inhaltsverzeichnisse beliebiger Disketten ausgeben lassen oder Programme aufrufen, die selbst kein Betriebssystem auf der Diskette enthalten. Mit einer weiteren Batch-Datei schalten Sie die RAM-Disk wieder ab. Nennen wir sie »ram/off« und geben nach dem Starten des Editors

```
ASSIGN c: df0:c
DELETE ram:c all quiet
```

ein. Nach dem Verlassen des Editors mit <ESC-x> kann die RAM-Disk nun mit »Execute ram/off« wieder vollkommen entleert werden. Sie beansprucht dann nur noch etwa 1 KByte Ihres Speichers. Die RAM-Disk selbst läßt sich nur durch einen Neustart des Systems beseitigen.

(Wilhelm Holtmeier/pa)

Datenübertragung mit dem Amiga

Ein Themengebiet, das sicher viele interessieren wird, ist die Datenübertragung vom und zum Amiga. Außer der reinen Übertragung von Daten gibt es aber auch noch andere Anwendungen.

Viele Besitzer eines Amiga haben sicher noch den »Vorgänger« (beispielsweise einen C 64) zu Hause stehen und wollen Daten von dem alten auf den neuen Computer übertragen. Der Weg solcher Daten führt dabei immer über die RS232-Schnittstelle (siehe Bild 1). Diese genormte Schnittstelle zur seriellen Übertragung gibt es auch am Amiga. Leider besitzt zum Beispiel der C 64 keinen solchen Anschluß, aber es gibt Interfaces, mit denen man an solchen Computer auch eine RS232-Schnittstelle anschließen kann. Das erste Problem ist das Kabel, mit dem die zwei Computer verbunden werden. Bei diesem Kabel (Bild 2) muß man aber auf einiges achten. Ein sehr wichtiger Punkt ist dabei die Abschirmung des Kabels, die erforderlich ist, wenn eine störungsfreie Übertragung gewährleistet sein soll. Dazu werden die beiden Anschlüsse (Pin 7) der zwei Stecker über die Abschirmung des Kabels miteinander verbunden.

Die zwei Anschlüsse TxD (Transmit Data = Serielle Sendeleitung) und RxD (Receive Date = Serielle Empfangsleitung) müssen über Kreuz verbunden werden. Dies kommt daher, daß die Sendeleitung des einen Computers auf die Empfangsleitung des anderen führen muß. In der anderen Richtung gilt dann natürlich das gleiche. Die anderen Anschlüsse, die hier verbunden sind, dienen nur dazu, daß der Computer auf alle Fälle sendet. Wenn das Kabel fertig ist, benötigt man nur noch die richtige Software zum Betreiben der Datenübertragung. Ein Programm (Convert 64) zur Konvertierung von Textdateien (zum Beispiel Vizawrite, Textomat) finden Sie in dieser Ausgabe. Damit ist die Übertragung vom C 64 zum Amiga vollständig problemlos. Schwierigkeiten gibt es noch, wenn Daten vom Amiga zu einem Computer mit ASCII-Zeichensatz geschickt werden sollen. Der Amiga benutzt nämlich andere Codes für die deutschen Umlaute als zum Beispiel der IBM-PC. Ein kleines Programm, das die deutschen Umlaute für den PC umsetzt, sehen Sie in Listing 1. Es soll bloß eine Anregung für diejenigen sein, die ein Konvertierungsprogramm schreiben wollen. Aufgerufen wird das Programm folgendermaßen:

convert <datei1> <datei2>

Es lädt zeichenweise die Daten aus der Quelldatei <datei1>, setzt für die Umlaute die richtigen Werte ein und speichert die Zeichen dann in <datei2>. Dabei kann man sich einen Vorteil des Amiga-Betriebssystems zunutze machen und für <datei2> einfach »ser:« schreiben. Dadurch werden die Zeichen gleich auf den seriellen Port des Amiga ausgegeben und können von dem anderen Computer mit einem DFÜ-Programm empfangen werden. Zu beachten ist dabei, daß die richtigen Parameter für die Datenübertragung mit dem Programm »Preferences« vorher eingestellt und gespeichert werden. Die Einstellungen kann man vornehmen, wenn man den Punkt »Change Serial« anwählt. Normalerweise benutzt man 8-Bit-Daten mit einem Stopbit, ohne Parität. Die Baudrate ist abhängig davon, wieviel der andere Computer schafft, da der Amiga auch hohe Raten bewältigt.

Eine weitere Anwendung der seriellen Kommunikation ist die Verwendung eines anderen Computers als Terminal am Amiga. Das heißt, daß mit dem zweiten Computer Befehle im Amiga ausgeführt werden. Dafür benötigt man jedoch den Gerätetreiber »aux:«, der im Directory »l« unter dem Namen »Aux-Handler« stehen muß, der zuerst für das System mit dem Befehl »mount aux:« verfügbar gemacht werden muß. Danach kann man mit »newcli aux:« ein neues CLI erzeugen, das seine Eingaben vom seriellen Port erhält. Der Benutzer am Terminal erhält allerdings nur alle Ausgaben, die in einem normalen CLI-Fenster erscheinen. Ruft der Benutzer also ein Programm auf, das Grafik darstellt, wird diese auf dem Bildschirm des Amiga gezeigt.

Bei diesem System kann man sogar Nachrichten an den Benutzer des Amiga schicken. Mit dem Befehl »copy * to con:<x0>/<y0>/<x1>/<y1>/titel« erscheint auf dem Amiga ein Fenster, bei dem die linke obere Ecke an den Koordinaten »x0« und »y0« liegt. »x1« und »y1« geben die Breite und Höhe des Fensters an und »titel« ist der Name des Fensters. Der Text, den man dann eintippt, erscheint in dem betreffenden CLI-Fenster auf dem Bildschirm des Amiga. Beendet wird dieser Vorgang durch Drücken von <CTRL C>.

Diese Kommunikation wäre jedoch sinnlos, wenn der Amiga-Benutzer nicht antworten könnte. Aber mit »copy * to aux:« können auch Nachrichten in die andere Richtung geschickt werden. Zum Beenden der Übertragung drückt man dann <CTRL \ >.

Wie man sieht, läßt sich mit der seriellen Schnittstelle des Amiga einiges anfangen. Die Datenübertragung ist also ein sehr interessantes Gebiet, von dem sicher noch oft in unserem Magazin die Rede sein wird.

Sollten Sie gute Programme für die Datenübertragung geschrieben haben (zum Beispiel ein Konvertierungsprogramm für C 64-Grafiken auf den Amiga), oder wenn Sie eine besondere Anwendung haben, schreiben Sie uns doch. (rb)

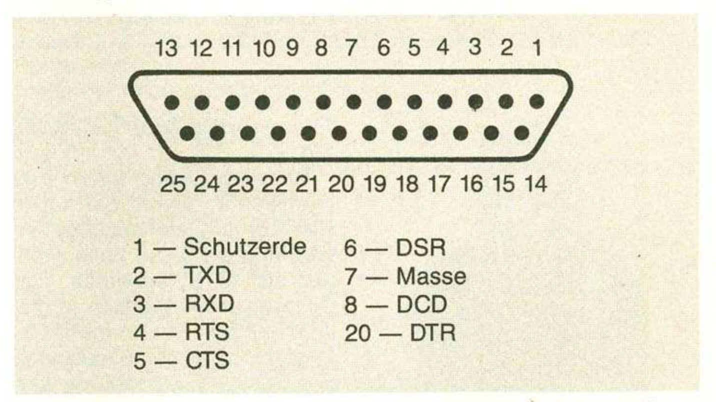


Bild 1. Die wichtigen Anschlüsse der RS232-Schnittstelle für die Datenübertragung zwischen zwei Computern

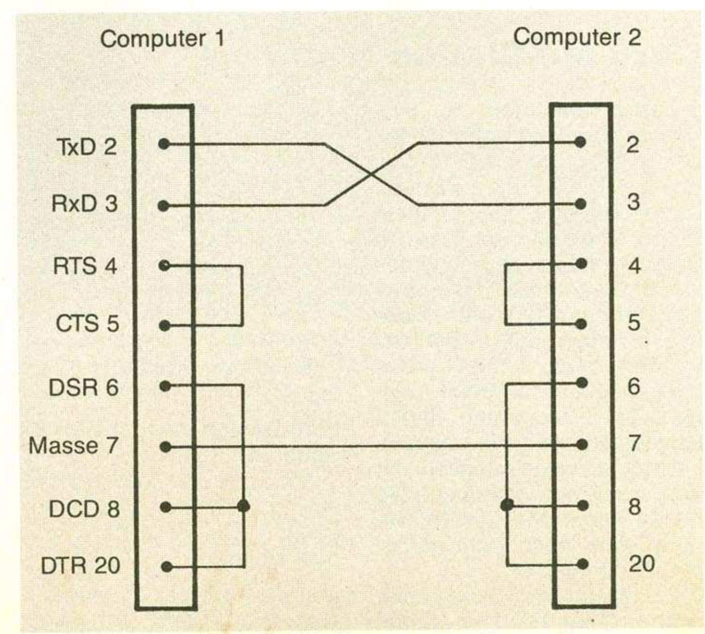


Bild 2. Ein Kabel zur Verbindung zweier Computer

```
#include <stdio.h>

main(argc,argv)
int    argc;
char   *argv  ;
{
    FILE   *fopen(),*fp1,*fp2;
    char   c,name 20 ;
    unsigned int zahl,anz;

    if(argc!=3)
    {
        printf(" nBenutzung: convert <datei1>
<datei2> n");
        exit(30);
        }

    if((fp1=fopen(argv 1 ,"r"))==NULL)
    {
        printf(" nDatei nicht gefunden! n");
        exit(30);
        }

    if((fp2=fopen(argv 2 ,"w"))==NULL)
    {
        printf(" nDatei kann nicht geoeffnet
        werden! n");
        exit(30);
        }
anz=0;
while((c=getc(fp1))!=EOF)
{
                if(c==' n')anz++;

                switch(c)
                {
                    case -4:
                        c=129;
                        break;
                    case -10:
                        c=148;
                        break;
                    case -28:
                        c=132;
                        break;
                    case -33:
                        c=225;
                        break;
                    case -36:
                        c=154;
                        break;
                    case -42:
                        c=153;
                        break;
                    case -60:
                        c=142;
                        break;


                putc(c,fp2);
                fclose(fp1);
                fclose(fp2);
                }
```

Listing 1. Convert: Ein kleines Konvertierungsprogramm



Guru Meditation

Jeder von Ihnen, der Programme schreibt, kennt sie leider zur Genüge: die Guru Meditation. Dieser rote blinkende Alarm bedeutet, daß irgendein Fehler aufgetreten ist. Da auch beim Schreiben und Abdruck von Programmen Fehler auftreten können, wird diese Ecke (wenn nötig) auch in den folgenden Heften auftauchen.

Wenn Sie also Fehler in einem der Programme finden sollten, schicken Sie uns eine Nachricht, wenn möglich mit Verbesserungsvorschlag. Sie helfen damit den Lesern, die den Fehler noch nicht gefunden haben.

Leider hat der »Guru« auch schon in unserer ersten Ausgabe zugeschlagen.

FiMon (Ausgabe 6/7 87, Seite 70)

Bei dem Listing treten Probleme auf, wenn der letzte Block einer Datei ausgegeben wird und die letzte Zeile weniger als 16 Byte enthält. Der Drucker erhält nämlich in diesem Fall kein Return mehr. Die folgenden Zeilen müssen nach der Zeile 493 eingefügt werden, um den Fehler zu beseitigen.

```
IF n> 0 THEN
  FOR i%=n TO 16
    PRINT #1,"   ";
  NEXT i%
  FOR i%=0 TO n-1
    wert=afeld%(i%)
    IF (wert> 32) AND
       (wert< 128) THEN
      PRINT #1,CHR$(wert);
    ELSE
      PRINT #1,".";
    END IF
  Next i%
  PRINT #1,CHR$(13)
END IF
```

Font-Design (Ausgabe 6/7 87, Seite 66)

Am Ende der Zeile 178 sind ein paar Zeichen verlorengegangen. Die Zeile muß komplett lauten:

```
b=-((ASC(MID$(a$,fa+y*bm+
(bmp+x)\8))AND 2^(7-(bmp+x)
MOD 8))> 0)
```

Die Clubecke

Viele unserer Leser entsprechen nicht dem gemeinhin dargestellten Typ von Computerfreak, der nächtelang vor seiner heißgeliebten Maschine hockt und sich für kaum etwas anderes interessiert als Bits und Bytes. Für die meisten ist die Computerei eigentlich eine Sache, die man am besten mit möglichst mehreren anderen gleichgesinnten Menschen betreibt. Dann macht sie auch erst richtig Spaß und man kann Ideen austauschen und zusammen spielen oder diskutieren. Im ganzen Land sprießen daher die Computerclubs nur so aus dem Boden und wer sich einmal für diese andere Seite der Computerei interessiert hat, wird die gewonnenen Kontakte zu schätzen wissen. Denn man hat unter anderem auch die Möglichkeit, eine Menge zu lernen und kann sich Fehler ersparen, die man vor allem als Einsteiger unausweichlich macht. Außerdem verfügen Computerclubs meist auch über die Möglichkeit, durch verschiedene Händler Hard- und Software billiger besorgen zu können. Natürlich tut sich auch auf dem Amiga-Bereich inzwischen etwas unter den User-Clubs. Wir bieten daher allen, die sich als Computerclub zusammengetan haben und hauptsächlich oder auch nur unter anderem mit der Amiga-Produktreihe arbeiten wollen, die Möglichkeit sich und ihren Club in unserer Clubecke vorzustellen. Es reicht im Prinzip eine kurze aber markante Beschreibung der Aktivitäten, Beiträge, Mitglieder oder geplanten Projekte mit denen man sich beschäftigt. Je knapper die Beschreibung ist (was nicht heißen soll, daß sie nicht sehr detailreich sein kann), um so besser stehen die Chancen, daß sie auch tatsächlich und im wesentlichen ungekürzt abgedruckt wird. Schreiben Sie uns und wir werden durch eine Auflistung in der Clubecke Ihnen den Kontakt zu anderen Clubs oder neuen Mitgliedern erleichtern. Es gibt sicher noch eine Menge Computerbenutzer, die zwar gerne solch einem Club beitreten würden, die aber bisher einfach noch nichts Richtiges gefunden haben.

Schreiben Sie an:
Markt & Technik Verlag AG
Amiga-Redaktion
— Stichwort Clubs —
z.H. Herrn Jörg Kähler
Hans-Pinsel-Str. 2
8013 Haar bei München

Simple Sprites in C

Auf einigen anderen Computern gibt es sie schon länger, die Sprites. Wie man die Simple Sprites auf dem Amiga benutzen kann, erfahren Sie hier.

Die Sprites werden durch die Hardware (Spritelogik im DENISE-Chip) über das vorhandene Bild gelegt, wodurch dem Benutzer Dinge wie Hintergrund oder Mischen mit dem Grafikspeicher abgenommen werden. Die Grafik-Bibliotheken des Amiga bieten einige Befehle, um mit solchen Sprites zu arbeiten. Doch zuvor noch etwas zum Aufbau derselben.

Es können maximal acht Sprites erzeugt werden, wobei Sprite 0 für den Mauszeiger reserviert ist. Die Breite eines Sprites ist auf höchstens 16 Punkte begrenzt, wohingegen die Höhe beliebig wählbar ist. Die Form eines Sprites wird zeilenweise beschrieben, wobei jede Zeile durch zwei Worte mit je 16 Bit dargestellt wird. Jedes Bit des ersten Wortes (High-Word) bestimmt mit dem entsprechenden Bit des zweiten Wortes (Low-Word) die Farbe eines Punktes, also das Farbregister, auf welches zugegriffen wird. Daraus ergeben sich also vier mögliche Farben für jeden Punkt, wovon eine die Hintergrundfarbe, also transparent, ist.

Hier ein Beispiel zur Beschreibung einer Spritezeile:

HIGH-WORD	LOW-WORD
0110110110110110	1001001001001001

Von den 32 Farbregistern des Amiga werden die Register 16 bis 31 für die Sprites verwendet. Es stehen also für die acht Sprites maximal 16 Farben zur Verfügung. Außerdem teilen sich zwei aufeinanderfolgende Sprites vier gleiche Farbregister, was folgende Aufteilung ergibt:

Sprite	Bitpaar	Farbregister	Sprite	Bitpaar	Farbregister
0/1	00	16	4/5	00	24
	01	17		01	25
	10	18		10	26
	11	19		11	27
2/3	00	20	6/7	00	28
	01	21		01	29
	10	22		10	30
	11	23		11	31

Wichtig ist es noch zu wissen, daß die Größe eines Spritepunktes unabhängig von der benutzten Auflösung ist, da ein Sprite immer in der niedrigsten Auflösung, also mit 320 x 256 Punkten, dargestellt wird.

Sprite holen mit GetSprite

Soweit zum Aufbau eines Sprites. Um nun ein Sprite in C aufzurufen, steht der Befehl GetSprite() zur Verfügung. Er hat folgende Syntax:

Sprite_Nummer=GetSprite(&sprite,nummer)

und liefert in der Variablen Sprite_Nummer bei erfolgreicher Zuweisung die Nummer des Sprites zurück, die in der Variablen »nummer« zugewiesen wurde. Ist das Sprite bereits zugewiesen oder nicht verfügbar, so wird -1 zurückgemeldet.

In nummer wird, wie schon erwähnt, die Nummer des Sprites beziehungsweise des verwendeten Spritekanals, übergeben. Gibt man hier -1 ein, so wird das nächste Sprite, beginnend mit 0, zugewiesen.

Die Variable »sprite« ist ein Pointer auf die Struktur des Sprites, die vor dem Aufruf initialisiert werden muß. Sie besteht aus der Form des Sprites, dem sogenannten Sprite-Image, so wie der Anzahl Zeilen, also der Spritehöhe.

Der Aufbau des Sprite-Images beginnt mit den SpriteControl-Words 1 und 2, die in einfachsten Falle beide Null sind, danach folgen paarweise das High- und Low-Word für jede Zeile, und endet mit zwei End-of-Data-Words, die wiederum Null enthalten.

Ist das Sprite nun erfolgreich aufgerufen, so ermöglicht der Befehl MoveSprite() das Bewegen desselben. Der Befehlsaufbau lautet

MoveSprite(vp,&sprite,x,y).

Hierbei ist »vp« ein Pointer auf die ViewPortStruktur, »sprite« wieder wie bei obigem Befehl der Pointer auf die Spritestruktur und »x« und »y« die Positionskoordinaten, die unser Sprite aufsuchen soll. Diese Koordinaten beziehen sich dabei immer auf die linke, obere Ecke des Sprites. Da die Sprites immer in der kleinsten Auflösung arbeiten, egal welche Auflösung der Hintergrund hat, haben wir eine Bildschirmbreite von 0 bis (320-Spritebreite), also 304 Punkte und eine Bildschirmhöhe von 0 bis (256-Spritehöhe) Punkte für den sichtbaren Bereich. Koordinaten außerhalb dieses Bereiches veranlassen das Sprite, den sichtbaren Bildschirmbereich zu verlassen.

Als nächstes ist es nun möglich, dem angewählten Sprite eine neue Form, also ein neues Sprite-Image zuzuweisen. Hierzu dient der Befehl ChangeSprite(). Er baut sich folgendermaßen auf:

ChangeSprite(vp,&sprite,neuimage)

Hierbei haben »vp« und »sprite« dieselbe Funktion wie oben, lediglich mit »neuimage« wird dem Sprite ein neues, vorher definiertes Sprite-Image zugewiesen.

Der letzte Befehl FreeSprite() ermöglicht es, das angewählte Sprite wieder abzuschalten. Mit

FreeSprite(nummer)

wird das Sprite mit der Nummer, die in der Variablen »nummer« übergeben wird, abgeschaltet.

Reichen die vier Farben eines Sprites nicht aus, so hat man auch die Möglichkeit, durch Verbinden von zwei Sprites die Farbauswahl auf 16 zu erhöhen. Diese Art von Sprite wird Attached Sprite genannt.

Hierbei müssen folgende Sprites kombiniert werden:

Sprite 1 mit Sprite 0	Sprite 3 mit Sprite 2
Sprite 5 mit Sprite 4	Sprite 7 mit Sprite 6

Die beiden Sprites müssen dann dieselbe Größe und Form haben und auch auf die gleiche Bildschirmposition gebracht werden, also übereinander liegen.

Durch Kombinationen der beiden Spritezeilen erhält man für jeden Punkt eine 4-Bit-Zahl, also 0 bis 15, die dann die Farbregister 16 bis 31 anwählt.

Folgende Farbregister werden somit zugewiesen:

Binär	Dezimal	Farbregister	
0000	0	16	(Hintergrundfarbe)
0001	1	17	
0010	2	18	
0011	3	19	
0100	4	20	
0101	5	21	
0110	6	22	
0111	7	23	
1000	8	24	
1001	9	25	
1010	10	26	
1011	11	27	
1100	12	28	
1101	13	29	
1110	14	30	
1111	15	31	

Als Beispiel für ein Attached Sprite durch Kombinationen von Sprite 3 mit Sprite 2 hätte man folgende Wörter zur Beschreibung des Images für eine Zeile:

Punktnummer	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Sprite3 High-Word:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Sprite3 Low-Word:	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
Sprite2 High-Word:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Sprite2 Low-Word:	0	0	0	0	1	1	0	0	0	0	1	1	0	0	0	0

Somit wird im obigen Beispiel der Punktnummer 6 die Bitkombination 0100 und somit das Farbregister 20 zugewiesen. Bei Punktnummer 10 ergibt sich die Kombination 0101 und damit Farbregister 26.

Um also ein Attached Sprite zu erhalten, definiert man die Images der beiden Sprites und ruft diese dann mit GetSprite() nacheinander auf. Danach muß man das SpriteAttachedBit in der Struktur des Sprites mit ungerader Spritenummer setzen. Dieses befindet sich im SpriteControlWord 2 an Bitposition 7; es wird also lediglich der Inhalt des SpriteControlWord 2 auf 128 (hexadezimal 80) gesetzt.

Sprites mit 16 Farben

Bewegt man danach die beiden Sprites mit MoveSprite() an die gleiche Bildschirmposition, so erscheint das Sprite mit 15 Farben und Hintergrundfarbe.

Zu erwähnen ist noch, daß normalerweise beim Einschalten des Amiga nur die Farbregister 16 bis 19 mit den Farben für die Workbench belegt sind. Die restlichen Farbregister enthalten Grauwerte. Um also farbige Sprites zu erhalten, müssen diese Farbregister vorher mit dem Befehl SetRGB4() mit den gewünschten Farbwerten gesetzt werden.

Zum Schluß noch ein einfaches Demoprogramm (siehe Listing 1), welches Simple Sprites erzeugt. Zuerst wird ein einfaches Sprite mit maximal vier Farben erzeugt, dessen Image dann verändert wird. Danach erscheint ein Attached Sprite mit 16 Farben.

(D. Kuntz/rb)

Programmname:	SimpleSprite
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	C
Compiler:	Lattice-C 3.03
Aufrufe:	lc -acbd simplesprite blink lib:c.o+simplesprite.o lib lib:lcm.lib+lib:lc.lib+lib: amiga.lib

```
Programm : sprite
-----------------

 1  /***************************************/
 2  /*       Ein Demo zum SimpleSprite      */
 3  /*                von                   */
 4  /*        D.Kuntz im April 1987         */
 5  /***************************************/
 6
 7  #include <exec/types.h>
 8  #include <graphics/sprite.h>      /* fuer Simple
    Sprite */
 9  #include <math.h>           /* fuer sin() */
10  #include <stdio.h>          /* fuer abs() */
11  #include <intuition/intuition.h>
12  #include <intuition/intuitionbase.h>
13  struct GfxBase        *GfxBase;
14  struct IntuitionBase *IntuitionBase;
15  struct Screen         *Screen;
16  struct ViewPort       *vp;
17
18  /******** Daten fuer SIMPLE SPRITE ********/
19
20  UWORD spriteimage_1 [] =  {
    /* Erstes Image fuer Sprite 1 */
21    0x0000,0x0000,       /* ControlWords 1+2 */
22    0x0c60,0x0fe0,       /*     Zeile 1      */
23    0x0380,0x1ff0,       /*           2      */
24    0x2380,0x3ff0,       /*           3      */
25    0x638c,0x7ffc,       /*           4      */
26    0x1c70,0x7ffc,       /*           5      */
27    0x1c70,0xfffe,       /*           6      */
28    0x1c70,0xfffe,       /*           7      */
29    0xe38e,0xfffe,       /*           8      */
30    0xe38e,0xfffe,       /*           9      */
31    0xe38e,0xfffe,       /*          10      */
32    0x1c70,0x7ffc,       /*          11      */
33    0x1c70,0x7ffc,       /*          12      */
34    0x1c70,0x3ff8,       /*          13      */
35    0x0380,0x1ff0,       /*          14      */
36    0x0380,0x0fe0,       /*          15      */
37    0x0000,0x0000    };  /*    End-of-Datas   */
38
39    short nummer = 2;
   /* Spritenummer des SimpleSprites */
40
41    struct SimpleSprite sprite = {
   /* Spritestruktur initialisieren            */
42      spriteimage_1,  /* Zeiger auf Spriteimage */
43        15            /* Anzahl Zeilen */
44      };
45
46      UWORD neuimage_1 [] =      {
   /* Zweites Image fuer Sprite1 */
47        0x0000,0x0000,        /* ControlWords 1+2 */
48        0x001b,0x0003,        /*     Zeile 1      */
49        0x0036,0x0c06,        /*           2      */
50        0x0036,0x1c06,        /*           3      */
51        0x006c,0x240c,        /*           4      */
52        0x006c,0x460c,        /*           5      */
53        0x00d8,0x7e18,        /*           6      */
54        0x00d8,0x4618,        /*           7      */
55        0x01b0,0x4630,        /*           8      */
56        0x01b0,0x0030,        /*           9      */
57        0x0360,0x0060,        /*          10      */
58        0xdb60,0x1860,        /*          11      */
59        0xeec0,0x08c0,        /*          12      */
60        0x77c0,0x03c0,        /*          13      */
61        0x3f80,0x0380,        /*          14      */
62        0x1980,0x0180,        /*          15      */
63        0x0000,0x0000    };   /*    End-of-Datas   */
64
65        /******** Daten fuer ATTACHED SPRITE ********/
66
67        UWORD spriteimage_3 [] = {
   /* Image fuer erstesSprite                    */
68          0x0000,0x0000,        /* ControlWords 1+2 */
69          0x0000,0xffff,        /*     Zeile 1      */
70          0x0000,0xffff,        /*           2      */
71          0xffff,0x0000,        /*           3      */
72          0xffff,0x0000,        /*           4      */
73          0xffff,0xffff,        /*           5      */
74          0xffff,0xffff,        /*           6      */
75          0x0000,0x0000,        /*           7      */
76          0x0000,0x0000,        /*           8      */
77          0x0000,0xffff,        /*           9      */
78          0x0000,0xffff,        /*          10      */
79          0xffff,0x0000,        /*          11      */
80          0xffff,0x0000,        /*          12      */
81          0xffff,0xffff,        /*          13      */
82          0xffff,0xffff,        /*          14      */
83          0x0000,0x0000,        /*          15      */
84          0x0000,0x0000,        /*          16      */
85          0x0000,0xffff,        /*          17      */
86          0x0000,0xffff,        /*          18      */
87          0xffff,0x0000,        /*          19      */
88          0xffff,0x0000,        /*          20      */
89          0xffff,0xffff,        /*          21      */
90          0xffff,0xffff,        /*          22      */
91          0x0000,0x0000,        /*          23      */
92          0x0000,0x0000,        /*          24      */
93          0x0000,0xffff,        /*          25      */
94          0x0000,0xffff,        /*          26      */
95          0xffff,0x0000,        /*          27      */
96          0xffff,0x0000,        /*          28      */
97          0xffff,0xffff,        /*          29      */
```

```
98          0xffff,0xffff,        /*         30      */
99          0x0000,0x0000  };    /*   End-of-Datas  */
100
101         short nummer_3 = 2;
     /*  Spritenummer des ersten Sprites         */
102
103         struct SimpleSprite sprite_3 = {
     /* Spritestruktur initialisieren            */
104           spriteimage_3, /* Zeiger auf Spriteimage */
105             30          /* Anzahl Zeilen */
106           };
107
108           UWORD spriteimage_4 [] = {
                /* Image fuer zweites Sprite              */
109             0x0000,0x0000,        /* ControlWords 1+2 */
110             0x0000,0x0000,        /*     Zeile 1      */
111             0x0000,0x0000,        /*           2      */
112             0x0000,0x0000,        /*           3      */
113             0x0000,0x0000,        /*           4      */
114             0x0000,0x0000,        /*           5      */
115             0x0000,0x0000,        /*           6      */
116             0x0000,0xffff,        /*           7      */
117             0x0000,0xffff,        /*           8      */
118             0x0000,0xffff,        /*           9      */
119             0x0000,0xffff,        /*          10      */
120             0x0000,0xffff,        /*          11      */
121             0x0000,0xffff,        /*          12      */
122             0x0000,0xffff,        /*          13      */
123             0x0000,0xffff,        /*          14      */
124             0xffff,0x0000,        /*          15      */
125             0xffff,0x0000,        /*          16      */
126             0xffff,0x0000,        /*          17      */
127             0xffff,0x0000,        /*          18      */
128             0xffff,0x0000,        /*          19      */
129             0xffff,0x0000,        /*          20      */
130             0xffff,0x0000,        /*          21      */
131             0xffff,0x0000,        /*          22      */
132             0xffff,0xffff,        /*          23      */
133             0xffff,0xffff,        /*          24      */
134             0xffff,0xffff,        /*          25      */
135             0xffff,0xffff,        /*          26      */
136             0xffff,0xffff,        /*          27      */
137             0xffff,0xffff,        /*          28      */
138             0xffff,0xffff,        /*          29      */
139             0xffff,0xffff,        /*          30      */
140             0x0000,0x0000  };     /*   End-of-Datas   */
141
142             short nummer_4 = 3;
                /* Spritenummer des zweitenSprites        */
143
144             struct SimpleSprite sprite_4 = {
                /* Spritestruktur initialisieren          */
145             spriteimage_4, /* Zeiger auf Spriteimage */
146                 30          /* Anzahl Zeilen */
147               };
148             short Sprite_Nummer;
                /* Rueckmeldung von GetSprite()           */
149
150 main()
151 {
152
153  short x,y;       /* Variablen fuer Sinusbewegung */
154  double yy;       /*          der Sprites          */
155
156  if (!(GfxBase = (struct GfxBase *)
157    OpenLibrary ("graphics.library",0)))
158      {
159        printf("Can't open Graphics-Library.\n");
160        exit(FALSE);
161      }
162
163      if (!(IntuitionBase = (struct IntuitionBase *)
164        OpenLibrary ("intuition.library",0)))
165          {
166            printf("Can't open Intuition-Library.\n");
167            CloseLibrary (GfxBase);
168            exit(FALSE);
169          }
170
171        Screen = IntuitionBase ->ActiveScreen;
172
173        printf("\f** SIMPLE SPRITE Demo von D.Kuntz
           **\n");
174
175        Sprite_Nummer = GetSprite(&sprite,nummer);
           /* Spriteaufruf */
176          if(Sprite_Nummer == -1)
             /* Sprite ansprechbar? */
177            {
178              printf("Kann Sprite nicht ansprechen.\n");
179              CloseLibrary (GfxBase);
180              CloseLibrary (IntuitionBase);
181              exit (FALSE);
182            }
183
184            SetRGB4 (&Screen ->ViewеLibrary
             (IntuitionBase);
181              exit (FALSE);
182            }
183
184            SetRGB4 (&Screen ->ViewPort,21,15,10,00);
             /* Farbregister setzen */
185            SetRGB4 (&Screen ->ViewPort,22,00,00,00);
186            SetRGB4 (&Screen ->ViewPort,23,15,00,00);
187
188            printf ("Ein SIMPLE SPRITE mit maximal
             4 Farben.\n");
189
190            for (x=0;x<321;x=x+2)
             /* Bewegung des Balls */
191              {
192                yy=abs(sin(x*2*PI/180)*180/PI);
193                y=185-yy*2;
194                MoveSprite (vp,&sprite,x,y);
195              }
196
197              ChangeSprite(vp,&sprite,neuimage_1);
               /* Neues SpriteImage */
198
199              printf("Das gleiche Sprite, aber mit
               einem neuem Image.\n");
200
201              for (x=321;x>-16;x=x-2)
               /* Bewegung des Amiga-Zeichens */
202                {
203                  yy=abs(sin(x*2*PI/180)*180/PI);
204                  y=185-yy*2;
205                  MoveSprite (vp,&sprite,x,y);
206                }
207
208                FreeSprite(nummer);
                 /*    Sprite loeschen */
209
210                /* Aufruf des Attached Sprites */
211
212                Sprite_Nummer = GetSprite
                 (&sprite_3,nummer_3);  /* Spriteaufruf */
213                  if(Sprite_Nummer == -1)
                   /* Sprite ansprechbar? */
214                    {
215                      printf("Kann Sprite nicht
                       ansprechen.\n");
216                      CloseLibrary (GfxBase);
217                      CloseLibrary (IntuitionBase);
218                      exit (FALSE);
```

Listing 1. Ein Demoprogramm für Simple Sprites (bitte ohne Zeilennummern eingeben)

```
219                         }
220
221                         Sprite_Nummer = GetSprite
    (&sprite_4,nummer_4);  /* Spriteaufruf */
222                         if(Sprite_Nummer == -1)
                            /* Sprite ansprechbar? */
223                           {
224                             printf("Kann Sprite nicht
                                ansprechen.\n");
225                             CloseLibrary (GfxBase);
226                             CloseLibrary (IntuitionBase);
227                             exit (FALSE);
228                           }
229
230                           SetRGB4 (&Screen ->
                              ViewPort,20,10,00,00);
                              /* Farbregister setzen */
231                           SetRGB4 (&Screen ->
                              ViewPort,21,00,10,10);
232                           SetRGB4 (&Screen ->
                              ViewPort,22,10,00,10);
233                           SetRGB4 (&Screen ->
                              ViewPort,23,10,10,00);
234                           SetRGB4 (&Screen ->
                              ViewPort,24,00,00,15);
235                           SetRGB4 (&Screen ->
                              ViewPort,25,15,00,00);
236                           SetRGB4 (&Screen ->
                              ViewPort,26,15,15,15);
237                           SetRGB4 (&Screen ->
                              ViewPort,27,15,00,15);
238                           SetRGB4 (&Screen ->
                              ViewPort,28,00,15,15);
239                           SetRGB4 (&Screen ->
                              ViewPort,29,15,15,00);
240                           SetRGB4 (&Screen ->
                              ViewPort,30,12,12,00);
241                           SetRGB4 (&Screen ->
                              ViewPort,31,00,15,00);
242
243                           spriteimage_4 [1] = 0x80;
                              /* SpriteAttachedBit setzen */
244
245                           printf("Nun ein ATTACHEDSPRITE
                              mit maximal 16 Farben.\n");
246
247                           for (x=0;x<321;x=x+2)
                              /* Bewegung des AttachedSprites
                               */
248                             {
249                               yy=abs(sin(x*2*PI/180)*180
/PI);
250                               y=150-yy*2;
251                               MoveSprite
                                  (vp,&sprite_3,x,y);
252                               MoveSprite
                                  (vp,&sprite_4,x,y);
253                             }
254
255                             FreeSprite(nummer_3);
                                /* Sprite loeschen */
256                             FreeSprite(nummer_4);
                                /* Sprite loeschen */
257
258                             CloseLibrary (GfxBase);
                                /* Libraries schliessen */
259                             CloseLibrary (IntuitionBase);
260 }

(c) M&T
```

Listing 1. Demoprogramm zu Simple Sprites (Schluß)

Hardcopy perfekt

Wer hat sich das nicht schon immer gewünscht, ein Hardcopy-Programm, das beliebige rechteckige Ausschnitte des aktuellen Bildschirms ausdruckt?

Mit »Hardcopy« wird dieser Traum Wirklichkeit. Die Bedienung von Hardcopy ist fantastisch einfach, obwohl das Programm wirklich sehr kurz geblieben ist. Zudem kann Hardcopy das ausgesuchte Rechteck auf dem Bildschirm auch noch vergrößert ausdrucken.

Doch zuerst kurz zu der Bedienung des Programms. Hardcopy kann auf zwei verschiedene Weisen gestartet werden, einmal durch Doppelklick auf dem entsprechenden Icon (Programmsymbol), oder durch Aufruf von einem CLI-Fenster mit »run hardcopy«. Bevor das Programm aber aufgerufen wird, muß mit dem Programm »Preferences« der richtige Druckertreiber eingestellt werden. Hardcopy benützt nämlich den voreingestellten Druckertreiber.

Hat man also das Programm gestartet, erscheint in der linken oberen Ecke des Workbench-Bildschirms ein kleines Fenster. Nach kurzem Warten erscheint dann ein Text in dem Fenster:
Bitte wählen ...
SHIFT P = PRINT
SHIFT L = LUPE
SHIFT X = ENDE

Dieses Fenster läßt sich verschieben und in der Größe verändern wie andere Fenster auch. Man bewegt jetzt das Fenster auf die gewünschte Position und stellt die erforderliche Größe ein, denn genau die Fläche, die unter dem Fenster liegt, wird später ausgedruckt. Drückt man dann <SHIFT p> erfolgt die Ausgabe des Bildschirmausschnitts in normaler Größe (Bild 1). Vor dem Drucken muß jedoch noch, der oben schon angesprochene, Druckertreiber geladen werden. Nach einiger Zeit ist dann der Ausdruck fertig.

Drückt man jedoch <SHIFT L>, wird ein vergrößerter Ausdruck des Ausschnitts ausgegeben. Die Größe der Grafik auf dem Papier ist dabei konstant, egal wie groß oder klein der eingestellte Bildschirmbereich ist. Der kleinste Bereich, der überhaupt ausgedruckt werden kann, ist auf dem Bildschirm ungefähr 5 mal 4 Zentimeter groß. Arbeitet man mit der Lupenfunktion, ist der Ausdruck dann so groß wie es geht, ohne die Seitenverhältnisse zu verändern.

Nach erfolgtem Ausdruck erscheint wieder dasselbe Fenster in der linken oberen Ecke. Will man keine weitere Hardcopy, drückt man einfach <SHIFT x> und beendet somit das Programm. Wenn Umstände wie nicht angeschlossener Drucker oder nicht vorhandener Druckertreiber den Ausdruck verhindern sollten, blinkt der Bildschirm ein paarmal und Hardcopy wird beendet.

Kurzes Listing — Große Hardcopy

Ein kleiner Wermutstropfen gehört aber doch zum Programm, es kann leider nur ausgedruckt werden, was sich auf dem Workbench-Bildschirm befindet. Man kann also keine Hardcopy von Bildschirmen (Screens) erstellen, die von einem anderen Programm extra geöffnet wurden. Aber vielleicht schreibt einer von Ihnen das Programm etwas um und schickt es uns zu.

Nun noch eine kleine Beschreibung des Programmlistings (Listing 1). Nach dem Einbinden der benötigten Headerfiles (Datei-

en mit Deklarationen für den C-Compiler) in den Zeilen 1 bis 5, geht es um die Definition der benutzten Variablen (Zeilen 7 bis 19).

Die erste Funktion des Programms (»openw«) eröffnet das Fenster, schreibt den Text hinein und wartet, bis eine der drei erlaubten Tastenkombinationen gedrückt wurde. Das Gegenstück dazu ist »closew«, es schließt den Port für den Printer und schließt die Bibliothek »IntuitionBase«, die vom Programm benötigt wird.

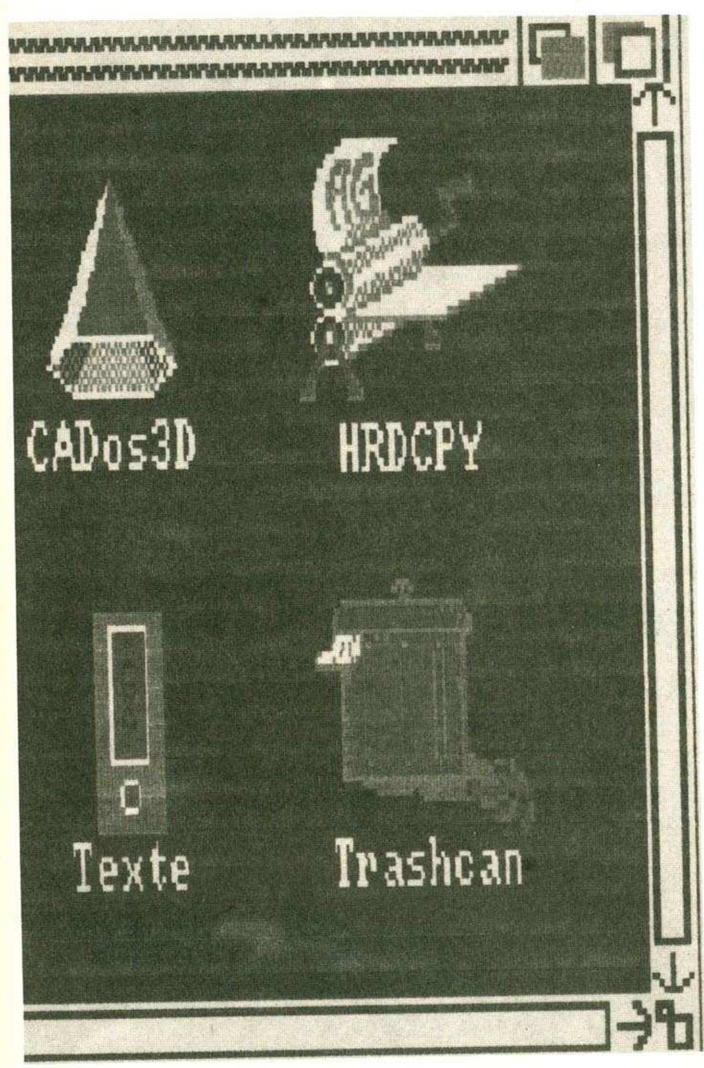


Bild 1. Ein Teil des Bildschirms als normale Hardcopy

Die eigentliche Ausdruckroutine ist »DumpRPort«, die im Prinzip nicht so schwer zu verstehen ist. Nach der Zuweisung der nötigen Werte wie Grafikmodus, Breite, Höhe und so weiter (siehe Zeilen 109 bis 119) wird der Druck einfach mit dem Befehl in Zeile 120 gestartet. Den Rest wie beispielsweise Umrechnen der Punkte erledigt das Betriebssystem.

»key« ist nur eine Hilfsfunktion, die einen Wert ungleich 0 zurückgibt, wenn eine Taste gedrückt wurde.

Das Hauptprogramm (»main«) eröffnet nur die benötigten Bibliotheken und den Port für den Drucker und ruft die Funktion DumpRPort auf. Danach wird nur noch der Drucker-Port geschlossen.

Wenn Sie dieses Programm verstanden haben, sollte es kein Problem mehr darstellen, Grafiken an einen Drucker zu schicken.

(Arno Gölzer/rb)

Programmname:	Hardcopy
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	C
Compiler:	Aztec-C Version 3.40A
Aufrufe:	cc -s hardcopy +l
Bemerkung:	In hardcopy.o -lc32

```
Programm : Hardcopy
-------------------

 1 #include <stdio.h>
 2 #include <exec/types.h>
 3 #include <devices/printer.h>
 4 #include <intuition/intuition.h>
 5 #include <libraries/dosextens.h>
 6 #include <intuition/intuitionbase.h>
 7 union printerIO *request; /* Zeiger auf Requeststruktur */
 8 union printerIO{
 9     struct IOStdReq ios;
10     struct IODRPReq iodrp;
11     struct IOPrtCmdReq iopc;
12     };
13
14 struct IntuitionBase *IntuitionBase;
15 struct FileHandle    *newwindow;
16 struct MsgPort            *printerPort;
17
18 int ledge,tedge,width,height,dest,spezial,flag;
19 int error=1;
20
21
22 /***          Funktionen          ***/
23
24
25 openw()
26 /* Fenster oeffnen und warten auf Tastendruck */
27 {
28     struct Window *aktiv, *hrdcpy_window;
29     char *zeichen;
30     char buffer[70];
31
32     sprintf(buffer,"RAW:0/0/160/80/HARDCOPY"); /* Window oeffnen */
33     if(!(newwindow=Open(buffer,MODE_NEWFILE))) return(1);/* Fehler */
34
35     aktiv=IntuitionBase->ActiveWindow;/* 'aktiv' = das neue Window */
36     if(flag==0) Delay(200);
37
38     /* Text ausgeben: */
39     sprintf(buffer,"%s","\033[7mBitte waehlen ...\033[0m\n\nSHIFT P = PRINT
       \nSHIFT L = LUPE\n\nSHIFT X = ENDE\n\n");
40     Write(newwindow,buffer,strlen(buffer));
41
42     /* Warten auf Shift L, -P oder -X :  */
43     while(zeichen!=0x4c && zeichen!=0x50 && zeichen!=0x58){
44             /* warten, bis HRDCPY-Window aktiv ist: */
45             while((hrdcpy_window=IntuitionBase->ActiveWindow)!=aktiv);
46             zeichen=key();
47     }
48     if(zeichen==0x58){                        /* Shift X */
49                     error=0;
50                     return(1);
51     }
52     flag=0;
53     ledge =hrdcpy_window->LeftEdge; /* Shift P oder -L */
54     tedge =hrdcpy_window->TopEdge;
55     width =hrdcpy_window->Width;
56     height=hrdcpy_window->Height;
57
58     Close(newwindow);                  /* Fenster schliessen */
59
60     if(zeichen==0x50){             /* Shift P */
61             dest   =width*11;      /* Groesse des Ausdruckes */
62             spezial=SPECIAL_MILCOLS | SPECIAL_ASPECT;
63     }
64     else{                          /* Shift L */
65             dest   =0;                               /* so gross wie  */
66             spezial=SPECIAL_FULLCOLS | SPECIAL_ASPECT;/* moeglich    */
67     }
68     return(0);
69 }
70
71
72 closew()
73 /* Programmende */
74 {
75     int i;
76
77     if(flag==1){
78             DeleteExtIO(request,sizeof(union printerIO));
79             DeletePort(printerPort);
80     }
81     if(error){
82             for(i=0;i<5;i++){
83                     DisplayBeep(0);
84                     Delay(5);
85             }
86     }
87     else Close(newwindow);
88
89     if(IntuitionBase) CloseLibrary(IntuitionBase);
90     exit(0);
```

Listing 1. Der sehr kurze Quellcode von Hardcopy

```
 91 }
 92
 93 DumpRPort(request)
 94 /* Raster von bestimmter Groesse auf Drucker ausgeben */
 95     union printerIO *request;
 96 {
 97     struct Screen   *screen;
 98     struct ViewPort *vp;
 99     struct RastPort *rp;
100     struct ColorMap *cm;
101     int mod;
102
103     screen  = IntuitionBase->ActiveScreen;
104     rp      = &screen-> RastPort;
105     vp      = &screen-> ViewPort;
106     cm      = vp     -> ColorMap;
107     mod     = vp     -> Modes;
108
109     request->iodrp.io_Command   =PRD_DUMPRPORT;
110     request->iodrp.io_RastPort  =rp;  /* Rasterport das augegeben wird*/
111     request->iodrp.io_ColorMap  =cm;  /* Zeiger auf ColorMapStructur */
112     request->iodrp.io_Modes     =mod; /* Hires, Interlace oder Sprites*/
113     request->iodrp.io_SrcX      =ledge;   /* linke Grenze        */
114     request->iodrp.io_SrcY      =tedge;        /* obere Grenze        */
115     request->iodrp.io_SrcWidth  =width;   /* Breite des Rasters  */
116     request->iodrp.io_SrcHeight=height;   /* Hoehe      des Rasters    */
117     request->iodrp.io_DestCols  =dest;    /* Format Spalte       */
118     request->iodrp.io_DestRows  =0;            /* Format Zeile        */
119     request->iodrp.io_Special   =spezial; /* Flags               */
120     return(DoIO(request));                /* DUMP ausfuehren     */
121 }
122
123 key()
124 /* Tastendruck */
125 {
126     char zeichen;
127
128     if(WaitForChar(newwindow,1)==0) return(0); /* keine Taste gedrueckt  */
129     Read(newwindow,&zeichen,1);        /* nach Tastendruck, dessen Wert in */
130     return(zeichen);                   /* 'zeichen' speichern              */
131 }
132
133 main()
134 {
135
136   if(!(IntuitionBase=(struct IntuitionBase *)
137      OpenLibrary("intuition.library",0))) closew();
138
139   for(;;){ /* Endlosschleife */
140
141     if(openw()) closew();   /* Programmende, wenn openw() 1 zurueckgibt */
142
143     flag=1; /* flag=1 bedeutet mindestens einen versuchten Ausdruck */
144     printerPort=CreatePort(0,0);     /* Printerport oeffnen */
145     /* mit Funktion CreateExtIO den Request initialisieren: */
146     request=(union printerIO *)
147               CreateExtIO(printerPort,sizeof(union printerIO));
148     /* Drucker Device oeffnen */
149     if(OpenDevice("printer.device",0,request,0)) closew();
150
151     /* Raster ausgeben */
152     if(DumpRPort(request)) closew();
153
154     /* Drucker schliessen */
155     CloseDevice(request);                             (c) M&T
156   }
157 }
```

Listing 1. Hardcopy (Schluß)

Speicheranzeige im CLI

Derjenige, der meistens im CLI arbeitet, wußte bisher nie, wieviel Speicherplatz ihm noch zur Verfügung stand. Das ändert sich jetzt mit »FreeMem«, einem kurzen C-Programm.

Wenn man einen Amiga mit 512 KByte besitzt, kann es schon mal passieren, daß man nicht genau weiß, ob der vorhandene Speicherplatz für ein weiteres Programm reicht. Es ist also sinnvoll, den noch vorhandenen Speicher ständig anzuzeigen.

Will man kein eigenes Fenster für die Anzeige aufmachen, bleibt nur noch ein Weg: die Darstellung in der Titelzeile des CLI-Fensters. In der Titelzeile steht ja normalerweise nur der Name des Fensters, beim CLI-Fenster »AmigaDOS«. Mit der Programmiersprache C kann man ohne größere Probleme auf diesen Namen zugreifen und ihn ändern. Der Name steht in der Struktur »Window« und hat die Bezeichnung »Title«. Diese Struktur ist im Betriebssystem des Amiga enthalten und findet sich beim Aztec-C-Compiler im Headerfile »include/intuition/intuition.h«.

Um nun den verfügbaren Speicher zu erhalten, muß man sich der Funktion »AvailMem« bedienen. Der Amiga verfügt über zwei verschiedene Speicherbereiche, das Chip- und das Fast-Memory. Die Spezialchips (Paula, Denise und Agnus) können nur auf das Chip-Memory zugreifen. Das Fast-Memory hingegen kann nur vom Prozessor benutzt werden, wodurch hier die Zugriffe schneller vonstatten gehen. Dies kommt daher, daß der Prozessor jederzeit auf den Speicher zugreifen kann, ohne auf die Spezialchips warten zu müssen. Aus diesem Grund wird bei FreeMem zweimal aufgerufen, da die Anzeige getrennt nach Chip- und Fast-Memory erfolgt.

Während FreeMem installiert ist, wird die Menüzeile des Bildschirms gesperrt. Das wird dadurch erreicht, daß FreeMem die Auswertung der rechten Maustaste übernimmt und die Ausgabe mit den Zeilen 71 bis 73 unterdrückt. Wer auf die Menüzeile nicht verzichten will, muß nur diese drei Zeilen löschen. Ausschalten läßt sich FreeMem durch doppeltes Klicken mit der rechten Maustaste in der linken oberen Fensterecke. Dies muß dann unbedingt getan werden, wenn das Fenster geschlossen werden soll, da sonst das Fenster auf dem Bildschirm stehenbleibt. Aber auch nach dem Befehl »endcli« kann man FreeMem noch ausschalten.

Einen Schönheitsfehler hat FreeMem aber doch, wenn man Programme wie die Metacomco Shell benutzt, die die Titelzeile schon zur Ausgabe von Informationen verwenden, funktioniert eine der Ausgaben nicht mehr. (Torsten Jürgeleit/rb)

Programmname:	FreeMem
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	C
Compiler:	Aztec-C Version 3.40A
Aufrufe:	cc freemem ln freemem -lc
Bemerkung:	Arbeitet auch mit Speichererweiterungen, Start mit run freemem

```
Programm : freemem
------------------

 1  /******************************************
 2   *  FreeMem  05/87  von Torsten Juergeleit  *
 3   *                                          *
 4   *  (fuer AztecC 3.4a und Kickstart 1.2)    *
 5   ******************************************/
 6
 7  #include <exec/exec.h>
 8  #include <intuition/intuition.h>
 9  #include <libraries/dosextens.h>
10  #include <devices/timer.h>
11  #include <functions.h>
12  #include <stdio.h>
13
14
15  #define REV            0L
16  #define EOS            '\0'
17  #define YES            TRUE
18  #define NO             FALSE
19  #define PRIORITY       -5L
20  #define SECONDS        0L
21  #define MICROSECONDS   500000L
22  #define SIG_TIMER      (1L << timer_port->mp_SigBit)
23  #define SIG_WINDOW     (1L << window->UserPort->mp_SigBit)
24
25  /* wichtige Deklarationen und Definitionen von globalen Variablen */
```

Listing 1. Der C-Quellcode von »Freemem« (bitte ohne Zeilennummern eingeben)

```
26
27 struct Process        *scd_proc;
28 struct MsgPort        *timer_port    = NULL;
29 struct IntuiMessage   *win_msg;
30 struct IntuitionBase  *IntuitionBase = NULL;
31 struct Window         *window        = NULL;
32 struct timerequest    timer_req;
33
34 struct MsgPort iorp = {{ 0, 0, NT_MSGPORT, 0, 0 }, 0, -1, 0,
35   { (struct Node *)&iorp.mp_MsgList.lh_Tail, 0,
36     (struct Node *)&iorp.mp_MsgList.lh_Head, 0, 0 }};
37 struct IOStdReq ior = {{{ 0, 0, 0, 0, 0 }, &iorp, 0 }, 0 };
38
39 BYTE  mem[60], new_mem[60];
40 BYTE  timer_name[20] = "proc_timer";
41 UBYTE *old_title     = NULL;
42 ULONG class, seconds, micros, last_sec = 0L, last_mic = 0L;
43 SHORT mouse_x, mouse_y;
44
45 BOOL init_proc(), create_timer(), double_click_exit();
46 VOID clean_up(), show_free_mem(), start_timer();
47 struct Window *get_window();
48
49  /* Haupt-Programm */
50
51 VOID
52 main()
53 {
54  BOOL  keepon = YES;
55  ULONG signals;
56
57  if (init_proc() == YES) {
58    ModifyIDCMP(window, window->IDCMPFlags | MENUVERIFY);
59    show_free_mem();
60    start_timer();
61
62    do {
63      signals = Wait(SIG_TIMER | SIG_WINDOW);
64      if (signals & SIG_TIMER) {
65        GetMsg(timer_port);
66        show_free_mem();
67        start_timer();
68      }
69      if (signals & SIG_WINDOW) {
70        win_msg = (struct IntuiMessage *)GetMsg(window->UserPort);
71        if ((class = win_msg->Class) == MENUVERIFY) {
72          win_msg->Code = MENUCANCEL;
73        }
74        seconds = win_msg->Seconds;
75        micros  = win_msg->Micros;
76        mouse_x = win_msg->MouseX;
77        mouse_y = win_msg->MouseY;
78        ReplyMsg(win_msg);
79        if (class == MENUVERIFY)  keepon = double_click_exit();
80      }
81    } while (keepon);
82    ModifyIDCMP(window, window->IDCMPFlags & ~MENUVERIFY);
83  }
84  clean_up();
85 }
86  /* Prozess initialisieren */
87
88 BOOL
89 init_proc()
90 {
91  IntuitionBase = (struct IntuitionBase *)
92    OpenLibrary("intuition.library", REV);
93  window    = get_window();
94  old_title = window->Title;                    /* alten Window-Namen retten
    */
95  if (! strncmp(old_title, " FreeMem!", 8)) {
96    printf("'FreeMem' already exists !\n");
97    return(NO);
98  }
99  if (! IntuitionBase || create_timer() == NO) {
100    printf("Failure while installing 'FreeMem' !\n");
101    return(NO);
102  }
103  window->Title = (UBYTE *)&mem[0];
104  printf("'FreeMem' installed !\n");
105  return(YES);
106 }
107  /* Beenden des Prozesses */
108
109 VOID
110 clean_up()
111 {
112  window->Title = old_title;
113  RefreshWindowFrame(window);
114  if (CheckIO(&timer_req.tr_node)) AbortIO(&timer_req.tr_node);
115  if (timer_req.tr_node.io_Message.mn_ReplyPort) CloseDevice(&timer_req)
    ;
116  if (timer_port) DeletePort(timer_port);
117  if (IntuitionBase) CloseLibrary(IntuitionBase);
118 }
119  /* Task mit dazugehoerigem Window suchen -         *
120   * entnommen aus 'SetFont 1.1' von Commodore-AMIGA */
121
122 struct Window *
123 get_window()
124 {
125  struct MsgPort        *con;
126  struct StandardPacket *packet = NULL;
127  struct InfoData       *id     = NULL;
128  struct Window         *w      = NULL;
129
130  if (OpenDevice("console.device", -1L, &ior, 0L)) clean_up(20);
131  if ((iorp.mp_SigBit = AllocSignal(-1L)) < 0) clean_up(30);
132  iorp.mp_SigTask = (struct Task *)FindTask(NULL);
133  if (iorp.mp_SigTask->tc_Node.ln_Type == NT_PROCESS) {
134    if (con = (struct MsgPort *)
135      ((struct Process *)iorp.mp_SigTask)->pr_ConsoleTask) {
136    if (packet = (struct StandardPacket *)
137      AllocMem((ULONG)sizeof(*packet), MEMF_CLEAR)) {
138    if (id = (struct InfoData *)
139      AllocMem((ULONG)sizeof(*id), MEMF_CLEAR)) {
140    packet->sp_Msg.mn_Node.ln_Name = (BYTE *)&(packet->sp_Pkt);
141    packet->sp_Pkt.dp_Link = (struct Message *)&(packet->sp_Pkt);
142    packet->sp_Pkt.dp_Port = &iorp;
143    packet->sp_Pkt.dp_Type = ACTION_DISK_INFO;
144    packet->sp_Pkt.dp_Arg1 = ((ULONG)id) >> 2;    /* BPTR !! */
145    PutMsg(con, packet);
146    WaitPort(&iorp);
147    w = (struct Window *)(id->id_VolumeNode);
148    FreeMem(id, (ULONG)sizeof(*id));
149  }
150  FreeMem(packet, (ULONG)sizeof(*packet));
151 }
152 }
153 }
154 ior.io_Unit = (struct Unit *) -1;
155 return(w);
156 }
157  /* Timer installieren */
158
159 BOOL
160 create_timer()
161 {
162  scd_proc = (struct Process *)FindTask(NULL);
163  sprintf(&timer_name[0], "fm_timer_port%d", (SHORT)scd_proc->pr_TaskNum
    );
164  timer_port = (struct MsgPort *)CreatePort(&timer_name[0], PRIORITY);
165  if (! timer_port || OpenDevice(TIMERNAME, UNIT_MICROHZ, &timer_req,
166    0L)) return(NO);
167  timer_req.tr_node.io_Message.mn_ReplyPort = timer_port;
168  timer_req.tr_node.io_Command              = TR_ADDREQUEST;
169  timer_req.tr_node.io_Flags                = 0;
170  timer_req.tr_node.io_Error                = 0;
171  return(YES);
172 }
173  /* Freien Speicher anzeigen */
174
175 VOID
176 show_free_mem()
177 {
178  ULONG chip, fast;
179
180  chip = AvailMem(MEMF_CHIP);
181  fast = AvailMem(MEMF_FAST);
182  sprintf(&new_mem[0]," FreeMem!  Chip: %-6ld  Fast: %-8ld  Total: %-8ld
    ",
183    chip, fast, chip + fast);
184  if (strcmp(&mem[0], &new_mem[0])) {
185    strcpy(&mem[0], &new_mem[0]);
186    RefreshWindowFrame(window);
187  }
188 }
189  /* Timer einstellen und starten */
190
191 VOID
192 start_timer()
193 {
194  timer_req.tr_time.tv_secs  = SECONDS;
195  timer_req.tr_time.tv_micro = MICROSECONDS;
196  SendIO(&timer_req.tr_node);
197 }
198  /* DoubleClick in linker, oberer Ecke ? */
199
200 BOOL
201 double_click_exit()
202 {
203  if (mouse_x >= 0 && mouse_x <= 10 && mouse_y >= 0 && mouse_y <= 10) {
204    if (DoubleClick(last_sec, last_mic, seconds, micros) == TRUE)
205      return(NO);
206    last_sec = seconds;
207    last_mic = micros;
208  }
209  return(YES);
210 }                                                      (c) M&T
```

Listing 1. Der C-Quellcode von FreeMem

Software-Marktübersicht

Ein wesentliches Kriterium für die Verbreitung und den Erfolg eines Computers ist das Angebot an Software, das erhältlich ist. Seit der Einführung des Amiga 1000 arbeiten verschiedene Software-Hersteller fleißig an allen Arten von Programmen für die Amiga-Serie.

Seit der letzten Marktübersicht in der CeBIT-Sonderausgabe hat sich vieles getan. Die Qualität der Software hat sich entscheidend verbessert. Auch das Anwendungsspektrum ist größer geworden. Man sieht, es tut sich was im Software-Sektor.

Die folgende Marktübersicht soll Ihnen eine Übersicht über die Amiga-Software geben. Um bei der Vielfalt des Angebots einen Überblick zu behalten, wurde die Tabelle in einzelne Bereiche eingeteilt.

Die Marktübersicht erhebt keinen Anspruch auf Vollständigkeit (Stand: Juni 1987). Es wurden nur Angaben übernommen, die wir bis zum Redaktionsschluß erhalten haben. Alle Einträge beruhen auf Hersteller- beziehungsweise Anbieterangaben. Einige Programminformationen lagen zum Zeitpunkt der Erstellung nur von amerikanischen Anbietern vor. Es ist nicht ausgeschlossen, daß sich bis zum Erscheinen dieser Marktübersicht ein deutscher Vertreiber gefunden beziehungsweise ein DM-Preis ergeben hat.

Die zum Teil sehr großen Preisunterschiede bei den einzelnen Programmen liegen darin begründet, daß sich mehrere Anbieter zum Vertrieb der Programme entschlossen haben (siehe Spalte »Anbieter«). Hier hilft nur ein direkter Preisvergleich durch Studium der verschiedenen Anzeigen, Anfordern von Preislisten, Katalogen und Prospektmaterial oder telefonische Nachfragen.

Die Amiga-Magazin-Marktübersicht macht keine Angaben zur Lieferbarkeit der aufgeführten Produkte, da sich hier die Aussagen von Hersteller und den diversen Anbietern oft erheblich unterscheiden. Es sind also die uns bekannten Programme, die 1987 bereits erhältlich sind oder noch erscheinen werden, in dieser Marktübersicht enthalten. Über Liefertermine informieren Sie sich bitte bei Ihrem Händler oder Distributor. (dm/jk)

Achtung! Bei Bestellungen aus dem Ausland (etwa USA) müssen zusätzlich noch Mehrwertsteuer und Zollgebühren aufgeschlagen werden.

Titel	Hersteller	ca. Preis in Mark $=Dollar	Anbieter
Programmiersprachen/Programmierhilfen			
A68020AMI (Assembler)		333	08
A68020MSAMI		398	08
AC Basic	Absoft	375-798	05,08,24,27,31,32,37
AC Fortran 77	Absoft	599-758	05,06,17,18,26,31,32,37
AC Fortran 77 /020	Absoft	1649	18
Amiga Programmers Lib.		489-628	06,17,18
Aztek C Commercial	Manx	1089-1320	05,06,15,17,18,26,31,37
Aztek C Developer	Manx	599-798	05,06,15,17,18,26,31,32,37
Aztek C Professional	Manx	445-690	05,15,17,26,31,35,37
Basic: P-Code Basic	Pecan Software	193	31
Basic Professional Pak	Pecan Software	386	31
C.A.D.S.	Future Computer		
Cambridge Lisp	Metacomco	390-650	04,05,06,13,15,17,18,26,31,35,37
Cross-Reference Generator		107	18
Cross-Assembler	Pecan Software	$79	
CSI Amiga Forth	Creative Solu.	448-578	26,37
DevPac	HiSoft	148	21
Fortran 77	Pecan Software	193	31
Fortran Professional Pak	Pecan Software	386	31
J-Forth	Delta Research	298	05,31
K-Seka Assembler	Kuma	139-248	02,04,32
Key to 'C'	DRP	148	15
Key to 'C' + dt. Handb.	DRP	148	05
Lattice C Compiler	Lattice	324-450	02,04,05,06,13,15,17,18,26,31,32,33,35,37
Lattice C Crosscompiler	Lattice	545-1298	05,15,17,18
Lattice dBC III Library	Lattice	299-917	08,18,31
Lattice LMK	Lattice		
Lattice MacLibrary	Lattice	199-275	18,31
Lattice Make Utility	Lattice	299	18
Lattice Panel	Lattice	388-409	05,15,18
Lint (C-SourceChecker)	Gimple-Soft	284	08
Make	Metacomco	158	05
Makro-Assembler	Commodore	$99	06
Makro-Assembler 68000	Metacomco	179-398	02,05,06,13,15,17,18,24,26,31,33,35,37
Modula 2 Commercial	TDI	619-848	02,05,06,15,17,18,24,26,31,37
Modula 2 Developer	TDI	325-505	02,05,06,15,17,18,24,26,31,32,37
Modula 2 Module Library	Pecan Software	288	08
Modula 2 P Code Modula 2	Pecan Software	$99	
Moduls 2 Professional Pak	Pecan Software	386	31
Modula 2 Progr. Tools	Pecan Software	237	08
Modula 2 Software Tools	Pecan Software	288	08
Modula 2 Standard	TDI	195-348	02,04,05,06,08,15,17,18,24,26,30,31,32,37
Multi-Forth	Creative Solut.	598	05,15
Pascal	Metacomco	185-450	02,04,05,06,13,15,17,18,24,26,31,32,35,37
Pascal Professional Pak	Pecan Software	386	31
PDQ Pascal	Pecan Software	212	08
Prolog	Philgerma	198	26

Titel	Hersteller	ca. Preis in Mark $=Dollar	Anbieter
Quelo 68000/010 Assembler	Quelo Inc.	249	18
Quelo 68020 Assembler	Quelo Inc.	369	18
Quelo Cross-Assembler MS-DOS/Amiga 68000 68010	Quelo Inc.	469	18
Quelo Cross-Assembler Amiga-Ext. Target	Quelo Inc.	1399	18
Quelo Cross-Assembler MS-DOS/Amiga 68020 68881	Quelo Inc.	585	18
Quelo Cross-Assembler Amiga-Ext. Target	Quelo Inc.	1775	18
Softworks Basic	Softworks		
True Basic	True Corp.	295-549	05,06,07,15,18,24,25,26,31,35,37
True Basic 3-D Graphic	True Corp.	105-135	05,06,15,25,31,32
True Basic Adv. Strings	True Corp.	105-135	05,06,31,32
True Basic Change	True Corp.	118	05,15
True Basic Chippendale	True Corp.	118	05,15
True Basic Developer	True Corp.	110-135	05,06,07,15,25,31
True Basic Library 1	True Corp.	110	18,25
True Basic Library 2	True Corp.	110	18,25
True Basic Library 3	True Corp.	110	18
True Basic Primer	True Corp.	$21	
True Basic Programs	True Corp.	$16	
True Basic Sort & Search	True Corp.	118-135	05,06,15,25,31
True Basic Stat	True Corp.	179-188	05,25,31
True Basic Runtime	True Corp.	328-399	05,06,15,25,31,35
UCSD Pascal	Pecan/TDI	189-398	02,08,17,31,32,37
Businessprogramme, Dateiverwaltungen und Kalkulation			
2+2 Home Budget Manager	Olamic Systems	119-199	06,18,31
2+2 Home Manager	Olamic Systems	119-219	06,18,31,32
Acquisition	Taurus Impex	649-799	06,31
ABase	Computer Power	$120	
Aegis Impact	Aegis	199-629	02,05,06,13,15,17,18,26,31,32,33,35,36,37
AMT (Amortisation)	The Other Guys	101	08
Amiga Scientific Calculator	Deskware	$20	
Analytic Art	Crystal Rose	$59	
Analyze 1.0	Micro Systems	229-395	02,04,06,08,15,17,18,26,32
Analyze 2.0	Micro Systems	330-428	05,06,18,31,37
Analyze Graph	Micro Systems	89	32
Best Business Management	Best Electronic	898	05,15,31
BTS Spreadsheet	Batteries Inc.	$69	
Bulletin Board	Micro Systems	229	31
Business and Experimental		475	18
Business Statistics		365	18
Calcraft	Synapse		
Calculator Construction Set	QuickSilver	$49	
Cash Register	Soft Touch	193	31
CCI Bottom Liner-A	Clockwork Comp.		
CCI Merchandiser-A	Clockwork Comp.		
CCI Super-A Merchandiser	Clockwork Comp.		
Computer Co-Pilot	Alive Systems G.	$900	
D.A.S. Home Finance	Advanced Soft.	$34	
Datamat	Data Becker	99	12
Datamat Relational DBMS		590	18
Datamat Relational DBMS stat.		825	18

Titel	Hersteller	ca. Preis	Anbieter
dBMan	Verasoft	319-548	05,06,15,17,18,24,26,31,32,37
Decision Analysis Techniques		275	18
Digital Building System	MicroMaster	$299	
Easy Checks	PAR Home	97	31
Easy Loans	PAR Home	78	31
Experimental Statistics		365	18
Exploratory Data Analysis		189	18
Financial Cookbook	Electronic Arts	99-158	05,06,15,18,32
Financial Plus	Byte by Byte	548-688	05,06,08,15,18,31
Flex File	Cardinal Soft.	$59	
Flow	New Horizons	193-271	05,06,08,13,15,18,24,31,37
Forcasting and Time-Series		365	18
Forecast & Marketing		495	18
FoxFile	Foxware	$50	
Grade Manager	Associated C.	174	31
Go Amiga Datei	Softwareland	199	31
Home Inventory Manager	Sunsmile	105	08
InfoBase	Harvsoft	$24	
Infominder		185	18
Investor	Diamond Soft.	1693	31
Isgur Portfolio	Batteries Inc.	$199	
Keep-Trak	The Other Guys	$129	
Lattice DBC II Lib.	Lattice	349	
Lattice Unicalc	Lattice	168-306	05,06,15,18,31
Linear Programming		239	18
Logistix	Grafox	249-578	02,05,06,14,15,17,18,26,31,32,35,37
LPD Filer	Digital Solut.	258-289	05,31
LPD Planner	Digital Solut.	258-289	05,31
MaxiDesk	Maxisoft	119-149	06,18,24,31,32
MaxiPlan	Maxisoft	289-299	06,18,31,32
MaxiPlan	Plus Maxisoft	385	31
MiAmiga File	Softwood	229-398	04,06,08,17,18,33
MiAmiga File II	Softwood	298	06
MiAmiga Ledger	Softwood	219-238	08,18
Money Mentor	Sedona Software	190-228	05,15,24,31
Multivariate Analyse		375	18
Nimbus 1	Nimbus	290	31
Omega File	The Other Guys	169-209	06,08,18,31
Optimization		275	18
Order	Micro Systems	119	08,31,32
Organize!	Micro Systems	209-271	05,06,08,15,18,31,32,37
Organizer	Northeast	119	31
Outline	PAR Home	97	31
PAR-Home 1 (FiBu)	PAR Home	97	31
PAR Home 2 (FiBu)	PAR Home	193	31
PAR Real 1 (Liegensch.)	PAR Home	288	31
PBS-Address	Raab & Co.	99	04
PBS-Data	Raab & Co.	99	04
PBS-Einnahme-Überschuß	Raab & Co.	198	04
PBS-Faktura	Raab & Co.	148	04
PBS-Lager	Raab & Co.	99	04
PERT & Critical Path		239	18
Phasar (Home Acc.)	Finally	209-228	05,18,31
Profimat	Data Becker	99	12
Promise	The Other Guys	119	31
Quality Control		365	18
Rags to Riches IV	Chang Labs	$499	
Record Master	Woodsoftware	148	05,15
Sales & Marketing		375	18
Shanner Planner	P.P.& S. Inc.	49	35
Softwood File II	Softwood C.	208-249	05,06,15,31,37
Softwood Ledger	Softwood C.	208	05,31
Station Manager	Associated C.	1843	31
Statistics for Marketing		375	18
Superbase	M&T	229-274	02,05,06,07,13,15,17,18,21,24,26,31,32,33,36,37
Superbase Personal	Progressive Per.	$149	
The Grid	TDI	97	31
The Shanner Planner	Shanner Int.	$39	
Trendrider	Finally	$99	
Video Manager	Sunsmile	$34	
VIP Professional	VIP Technologies	298-798	02,05,06,15,17,18,24,26,30,31,32,37
Textverarbeitung			
Beckertext	Data Becker	a.A.	12
City Desk	Run Size	325	31,35
DesignText	DesignTech	248	08
Dynamic Word	Micro Illus.	290	31
Go Amiga Text	Softwareland	199	31
LaserScript	Gold Disk	85-99	31,37
LPD Writer	Digital Solut.	258-289	05,31
Mastertype	Mindscape	79	18,31,32

Titel	Hersteller	ca. Preis	Anbieter
PageSetter	Gold Disk	248-429	02,05,06,13,15,17,24,26,30,31,32,37
PageSetter Professional	Gold Disk	765	31
PaperClip Elite	Batteries Inc.	$129	
Professional Text Engine	Zirkonics	$99	
ProWrite	New Horizons	198-398	05,06,08,15,30,31,37
Publisher 1000	Northeast	389-459	05,15,31,32,37
Scribble	Micro Systems	198-330	08,17,18,24,26,31,32
Scribble Plus	Micro Systems	219-235	05,06,15,37
Softwood Writer	Softwood	193	31
Talker	Finally Soft.	148-185	05,06,15,31
Textcraft	Commodore	85-99	06,18
Textcraft Plus	Arktronics	89	
Textmaster	Haitex	$79	
Textomat	Data Becker	99	12
TxEd	Microsmiths	88-128	02,05,08,13,15,31
UBM Text V2.2	UBM-Drecker	228-249	05,06,15,17,24,26,34,35,37
Vizawrite	Viza-Software	198-498	02,04,05,15,31,32,37
Word Perfect	Wordperfect Co.	763	31
WriteHand	Byte by Byte	119	31
CAD/CAE/CAM, Animations- und Grafiksoftware			
Aegis Animator/Images	Aegis	239-398	05,06,13,15,17,18,24,26,30,31,32,35,36,37
Aegis Art Pak 1	Aegis	65-89	05,06,13,15,17,18,31,32,35
Aegis Art Pak 2	Aegis	89	05
Aegis Draw	Aegis	249-598	02,05,06,13,15,17,18,31,32,33,37
Aegis Draw Plus	Aegis	419-615	02,05,06,07,13,15,17,18,24,26,31,32,33,35,37
Aegis Images	Aegis	69-175	05,06,13,17,18,31,32,33,35
AmigaPlot V1.0	Tychon Tech.	$49	
Analytic Art	Crystal Rose	116	31
BobShop (+Soundshop)	Revolution Soft.	$49	
Brush Works	Associated C.	79	31
Butcher	Eagle Tree	72-99	06,16,21,31,37
Cale Fonts	Kreativ Teknikk		
Calligrapher amerik.	Interactive S.	148-219	05,15,21,31,37
Calligrapher dt.	Interactive S.	269	21
Coloring Book 1	TDG	88	05,06,15
Coloring Book 2	TDG	88	05,15
Coloring Book 3	TDG	88	05,15
Degas Elite	Batteries Inc.	159-189	32
Deluxe Help	Electronic Arts	49-58	05,15,35
Deluxe Paint	Electronic Arts	219-249	02,04,06,07,11,18,26,32,36
Deluxe Paint II	Electronic Arts	249-299	02,05,04,06,07,13,15,17,18,21,24,26,30,31,32,33,35,36,37
Deluxe Paint II mit dt. Anwenderhandb.	Electronic Arts	278	05,31
Deluxe Paint Data Disk 1	Electronic Arts	53-69	05,06,07,15,18,31,35
Deluxe Paint Data Disk 2	Electronic Arts	69	24,31,32,37
Deluxe Paint Data Disk 3	Electronic Arts	69	31
Deluxe Print	Electronic Arts	139-249	02,04,05,06,07,13,15,17,18,21,26,31,32,33,35,36,37
Deluxe Print Art Disk #1	Electronic Arts	67-69	05,06,07,13,15,17,18,35,37
Deluxe Print Art Disk #2	Electronic Arts	57-69	18,32,36
Deluxe Video 1.0	Electronic Arts	188-298	02,04,05,06,07,11,13,15,18,21,24,26,30,32,33,36
Deluxe Video 1.2	Electronic Arts	248	05,31,36,37
DesktopArtist	Sunrize Ind.	58	31
Digi Paint	New-Tek	118	05,15,31
Digi-Pix	DIP	78	05,15
Dynamic CAD	Micro Illus.	978-1298	04,05,06,15,17,18,24,26,31,37
Express Paint	PAR Home	193	31
Graphicraft	Commodore	82-89	06,11,18
Logic Works	Capilano Syst.	239	18,31
Newio	Alphatron	498	02,15,26
Newio Developer	Alphatron	998	02
New Tech Coloring Book	Electronic Arts	45	05,06,32
PCLO Platinen-Layout	Softcircuits	2919	18
Print Master	Unison World	109-119	05,07,10,15,18,24,31,32
Print Master Art Gallery 1	Unison World	59-69	05,07,15,18,31,32
Print Master Art Gallery 2	Unison World	69	05,07,18,31
Prism	Impulse	159	21,31,37
ProClip	Impulse	58	31
Pro Video Titel Generator	JDK Images	178-398	05,08,15,26
Raster Pic./Pin.	Melzer		22
SuperWriter dt.	Atlantis	98	05

Titel	Hersteller	ca. Preis	Anbieter
Talking Coloring Book	JMH Software	$29	
TV Text	Zuma Group	249-579	05,06,13,15,31,37
Video Draw 1		79	07
Video Wizard	Intracorp	99	32
Musiksoftware			
Aegis Sonix 2.0	Aegis	139-289	02,05,06,07,13,15,17,18,24,30,31,32,35,37
Deluxe Music	Electronic Arts	159-249	02,05,06,10,13,15,17,18,21,24,26,31,32,33,35,36,37
Deluxe Music 2.0	Electronic Arts	215	18,31
Deluxe Music Data 1	Electronic Arts	69	31
DNA & Fractal-Music	Silver Software	$19	
E.C.E Midi	Midi Interfaces	188	05
Future Sound Drum Kit	Applied Visions	395-409	05,06,15,17,31,37
Instant Music	Electronic Arts	71-198	04,05,06,10,13,15,17,18,24,26,32,33,35,37
Instant Music Data 1	Electronic Arts	59-68	05,06,15,18,32,37
Midi Gold	Midi Interfaces	188	05,06,31
Music Student 1	Associated C.	$59	
Music Studio	Activision	88-129	01,02,03,04,05,06,15,17,18,26,32,35
Music X	Micro Illus.	572	31
Perfect Sound	Run Size Ind.	198	05,31
Pro Midi Studio	Mimetics	179-490	05,06,07,10,15,17,18,23,24,31,37
Sound Effects Library	Karl R. Denton	$99	
Soundsampler	Mimetics	219-225	24,37
Sonix Dreams 1	Atlantis	58	05
Sonix Dreams 2	Atlantis	58	05
Studio Magic	Sunrize Ind.	129	31,35
Waveform Easel	Silver Software	$19	
XPress Patch	Digitools	$24	
Datenfernübertragung			
A-Talk	Felsina Soft.	169	08
A-Talk Plus		323	08
A-Term		105	24
Amiga Kermit: SourceCode	TDI	58	31
AmigaTerm	Commodore	99	06,18,31
Aegis Diga!	Aegis	139-189	06,31,35
BBS-PC	Micro Systems	210-271	05,06,08,15,18,24,37
Casa Mi Amiga BBS	Micro Systems	$99	
Digital Link	Digital Creations	148-219	05,06,15,18,31,37
Genie	Data Dynamics		
Macromodem	Kent Engineer.	158	05,15,31
MaxiComm V3.6	Maxi Soft	105-147	06,18,24,31
MiddleMan	Benaiah Comp.	$59	
Online!	Micro Systems	148-330	02,05,06,15,17,24,26,28,31,37
Teleport II	Cherokee Data	$99	
TY-100 V1.2	Tychon Tech.	$69	
Tools, Utilities und Zusätze			
A-Copier	Megasoft	$35	15
A-Disk	Megasoft		
A-Edit	DRM	198	05
A-Filer	Megasoft	105	24
A-Mon dt.	Atlantis	138	05
ADFO (File Organizer)	Westcom Ind.	119	31
Amiga Editor	TDI	97	31
Boot-Boy dt.	Atlantis	68	05
C-Monitor	Diamond Soft.	195-199	04,07,17,31,37
CLI-Mate	Prog. Periph.	65-78	05,06,15,21,24,31,32,37
Clip	ECE Research	126	31
Copy II	Central-Point	$39	
Custom Screens	Soft Touch	135	31
D'Buddy Hardcopy	Digital Creat.	178-189	05,15,31
Dimmer	PAR Home	39	31
Discovery Shell	Discovery	119	16
Disk 2 Disk	Central Coast	119	31
Disk 2 Disk + dt. Anwenderbuch	Central Coast	98	05
Disk Library	Classic Image	109	32
Disk Pro Plus	PAR Home	58	31
Diskwik	Tigress	128	13,15
Diskwik + dt. Handbuch	Tigress	108	05
DiskWorks	Discovery	199	16
DOS 2 DOS (ST -> Amiga)	Central Coast	128	05,31
DOS Express	Bantam Soft.	60	15,24,31
Drive Alignment Kit	ECE Research	578	31
DX 16	Discovery	119	16
Encore	Oxxi Inc.	76	31
Expert System Kit	Interactive A.	$69	
Explorer Monitor & Tool	Interactive A.	99	31,35
FastFonts	Microsmiths	89-139	08,31
FlipSide	Micro Systems	116	31
Gizmoz Produktivity Set	Digital Creat.	109-248	02,05,06,15,17,18,26,31,32,37
Go-Encrypt	DataTalk Inf.	$29	
Gold Spell	Gold Disk	98	05,15,31
Grabbit	Discovery	65-129	05,06,15,16,17,18,31,37
HardHat	Westcom Ind.	159	31
HD 2 Disk	C Ltd.		16
Keep Trak	The Other Guys	$129	
Key Genie	Discovery	118	05,16,31
Laser Writer Fonts Vol.1		99	18
Laser Writer Utilities Vol.1		99	18
Lattice Screen Editor	Lattice	198-275	05,15,18
Lattice Text Utilities	Lattice	169-175	18
Marauder 2	Discovery	80-139	05,06,07,13,15,16,18,24,31
MaxiCache	Maxi Soft	59	18
MaxiKey	Maxi Soft	59	18
MaxiMizer	Maxi Soft	59	18
MaxiShare	Maxi Soft	59	18
Megakick	M. Lamm	39	09
Menu Maker	Pecan Software	$49	
Metacomco Shell	Metacomco	118-169	05,06,13,26,31,32,33,37
Metacomco Toolkit	Metacomco	99-139	05,13,15,17,26,31,32,33,37
Metascope (Debugger)	Metadigm Inc.	$95	
Metascribe (Editor)	Metadigm Inc.	$85	
Metatools	Metadigm Inc.	$69	
Micro-Shell	Meta-Soft	$59	
Mirror Copier	Compumed	98-152	05,06,07,08,13,15,24,31
Mirror Hacker Package	Compumed	98-152	05,06,08,13,15,31
Par Home 1	Par Software	158	05
Par Real 1	Par Software	498	05
Peggy	SAS	88	29
Perfect Score	Mindscape	179	31
PowerPak	Intuitive Tech.	$99	
Power Windows	Inovatronics	198-209	05,13,15,31
Quick Nibble	Copperstate	118	05,15,31
SmartDisk	Know Technology	a.A.	05,15
T-Util	TechniSoft	$49	
Tree Drive	Intelogic		
Transformer (MS-DOS)		170	11
Tychon Utilities 1.1	Tychon Tech.	$49	
White Lightning		a.A.	15
Zing!	Meridian Soft.	179-228	05,06,13,15,17,21,31,37
Zing!	Keys Meridian Soft.	109	21,31
Zuma Fonts 1	Zuma Group	68-128	05,06,08,13,15,18,31,32,37
Zuma Fonts 2	Zuma Group	68-128	05,06,08,13,15,18,31,32,37
Zuma Fonts 3	Zuma Group	68-128	05,06,08,13,15,18,31,32,37
Spiele			
Aldaron	Microshiths	$39	
Alien fires	Jaghware Inc.	84-95	05,15,24,31,32,37
Archon	Electronic Arts	62-99	02,04,06,10,15,17,18,24,26,28,32,35,37
Archon II Adept	Electronic Arts	59-110	04,05,06,10,17,18,24,26,28,32,35,37
Arcticfox	Electronic Arts	62-129	04,05,06,10,15,17,18,24,31,32,35,37
Arena	Psygnosis	75-119	03,04,06,10,17,24,26,28,31,32,35,37
Autoduell	Electronic Arts	89-119	05,15,31,32
Bad Cat	Rushware	a.A.	28
Balance of Power	Mindscape	79-119	04,05,06,10,15,24,28,31,32,36,37
Baseball Gamestar			
Basketball	Activision	80	01
BMX-Trials	Kingsoft	30	19
Break	Diamond Softw.		
Bridge 4.0	Artworx	59-72	18,31,32
Bridge 5.0	Artworx	68	31
Challenger	Kingsoft	30	19
Championship Baseball	Activision	79-98	05,31,37
Championship Basketball	Activision	79-88	05,06,15,31
Championship Football	Activision	79-89	10,15,31,37
Championship Golf	Activision	79-98	03,05,06,10,15,31,32,37
Chessmaster 2000	Electronic Arts	88-149	02,05,06,10,13,15,17,18,24,26,31,32,33,35,37
Chessmate	Dark Horse	58-65	06,18,31
City Defence	Kingsoft	30	07,19

DIE FASZINATION



7/8-'87 DAS GROSSE PERSONALCOMPUTER-MAGAZIN

PCs, die man sich leisten kann
- ★ Wer ist der Beste? Commodore PC10, Schneider PC, Sanyo PC16 Plus, Zenith Z-148
- ★ Große Marktübersicht

Riesen Programmierer-Teil
- ★ Super-Listings
- ★ Tips&Tricks

Große Leistung super preiswert
- ★ IBM Modell 30
- ★ Atari PC-1/PC-2
- ★ Commodore PC-1

Software-Tests
- ★ Farbenfroh: Paintbrush
- ★ Schnell: Turbo Basic
- ★ Duell: GEM kontra Windows

ERSTAUSGABE

Sichern Sie sich jetzt Ihre persönliche PC-Zukunft.

Unermeßliche Bereiche einer faszinierenden Computerwelt entdecken Sie durch das PC-Magazin-Plus. Sie lernen eine ganz neue Welt der IBM-PCs und kompatibler Systeme mit überraschenden Perspektiven und Möglichkeiten kennen – beim Programmieren, bei Text- und Datenverarbeitung, Grafik und Homeentertainment.
PC-Magazin-Plus, die ganz neue PC-Zeitschrift für alle, die IBM-PCs und Kompatible mit Engagement benutzen oder einsetzen wollen – ob Einsteiger oder Profi. Sie erhalten sie im Abonnement oder bei Ihrem Zeitschriftenhändler.

Kennenlern-Angebot

mit einem kostenlosen Probeexemplar PC-Magazin-Plus

Ja, ich interessiere mich für PC-Magazin-Plus und möchte ein kostenloses Probeexemplar dieser Zeitschrift. Wenn ich PC-Magazin-Plus weiterlesen will, brauche ich nichts zu tun, ich bekomme dann PC-Magazin-Plus regelmäßig per Post zum günstigen Jahrespreis von 84,— DM (für 12 Ausgaben, Auslandspreise und Studentenabo siehe Impressum).

Geld-zurück-Garantie:

Ich kann das Abonnement jederzeit kündigen, es gibt keine Kündigungsfrist. Zuviel bezahlte Beträge erhalte ich zurück.

Name ____________ Vorname ____________

Straße ____________ PLZ/Wohnort ____________

Telefon ____________ Datum, 1. Unterschrift ____________

Mir ist bekannt, daß ich diese Bestellung innerhalb von 8 Tagen bei der Bestelladresse Markt & Technik Verlag Aktiengesellschaft, Hans-Pinsel-Str. 2, 8013 Haar bei München widerrufen kann. Zur Wahrung der Frist genügt die rechtzeitige Absendung des Widerrufs an obige Adresse. Ich bestätige dies durch meine 2. Unterschrift.

Datum, 2. Unterschrift ____________

HCS18 A 8/9/87

Titel	Hersteller	ca. Preis	Anbieter
Computer Baseball	Strategic Sim.	69-79	31,32
Cosmos	HC Software	$38	
Cruncher Factory	Kingsoft	30	19,32
Deep Space	Psygnosis	75-129	02,03,04,06,10,17, 24,26,28,32,35,37
Defender of the Crown	Cinemaware	88-198	02,04,05,06,07,10,15, 17,18,24,26,28,31, 32,35,36,37
Delta Patrol	Other Valley	79	18
Demolition	Kingsoft	30	19,32
Diablo	Classic Image	79	06,18,31,32
Donald Ducks Playground	Sierra Online		
Emerald Mine	Kingsoft	30	19
F-15 Strike Force	Micropose	78	31
Financial Time Machine	Insight	69-99	24,32
Fire Power	Micro Illus.	49	31
Flight II	Sublogic	95-178	02,03,04,05,06,07,10, 13,15,17,18,24,26,28, 30,31,32,35,36,37
Flight II Scenario Disk 1	Sublogic	55	04,13,24,37
Flight II Scenario Disk 2	Sublogic	55	31
Flip Flop	Kingsoft	30	19,28,32,35,37
Football	Gamestar		
Fortress Underground	Kingsoft	30	19
Gato	Spectrum Holo- byte	89	06,18,32
Gokart Racing	Kingsoft	30	19
Golden Oldies	Electronic Arts	69-85	06,18,32
Grand Master Chess	Sierra Online	79	32
Grand Prix	Electronic Arts	89	04,32
Grand Slam Tennis	Infinity Soft.	79-148	05,06,15,24,26,31, 32,37
Gridirion	Bethesda Soft.	135-158	05,15,24,31
Gridirion ProData Disk	Bethesda Soft.	$19	
Gunship	Micropose	88	05,31,32
Hacker 1	Activision	59-99	01,02,03,04,05,06,10, 15,17,18,26,31,32,35, 37
Hacker 2	Activision	62-99	01,02,03,04,05,06,10, 15,17,18,32,35,37
Halley Project	Mindscape	69-99	02,05,06,10,15,17,18, 24,26,28,31,32,35, 36,37
Hex	Mark o/t Unicorn	79	24,32
Hole in one Golf	Artworx	59	32
Hollywood Poker	Rushware	49	24,28
Kampfgruppe	SSI	140	07,15,24,31
Karate King	Kingsoft	50	19,32,37
Karate Master	Rushware	a.A.	28
Keyboard Cadet	Mindscape	79-99	06,17,18,31,32
King of Chicago	Mindscape	99-108	04,05,31,32,35,37
Las Vegas	Kingsoft	30	19
Leader Board	Access	62-99	04,05,06,07,10,15,17, 18,24,26,28,32,35,37
Leaderboard Tournament	Access	38-69	04,05,06,15,17,24, 28,32,35,37
Little Computer People	Activision	69-129	02,03,04,05,06,10,15, 17,18,26,28,32,35,37
Marble Madness	Electronic Arts	61-109	02,04,05,06,10,15,17, 18,24,26,28,31,32, 35,37
Mean 18	Accolade	78-99	05,06,15,18,24,28,31, 32,36
Mean 18 Coursedisk	Accolade	49	32
Micro League Baseball	Micro League	$59	
Mindbreaker	Rushware	a.A.	28
Mindwalker	Synapse	105	06,18
Monkey Business	Other Valley	99	10
Ogre	Electronic Arts	79-118	05,06,07,15,24,31
One On One	Electronic Arts	75-99	02,04,05,06,10,15,17, 18,24,26,28,32,35,37
Pac Boy	Rushware	a.A.	28
Peggammon	Kingsoft	30	19
Phantasie	SSI	79	31,32
Phalanx	Kingsoft	30	19,32
Portal	Activision	89-129	01,02,03,04,05,06,10, 15,17,24,26,31,32, 35,36,37
Quintette	Miles Computing	89-99	05,06,15,24,31,32
Quiwi	Kingsoft	49	19,26,28,32,35,37
Return to Atlantis	Electronic Arts	89-129	04,05,06,15,17,32
Roadwar 2000	SSI	78	31
Rocket Attack	Rushware	a.A.	28
S.D.I.	Cinemaware	88-119	02,04,05,06,07,13,15, 24,28,31,32,35
Shanghai	Activision	59-99	01,02,03,04,05,06,07, 10,15,17,18,24,31,32, 35,36,37
Shooting Star	Rushware	a.A.	28

Titel	Hersteller	ca. Preis	Anbieter
Silent Service	Micropose	79-90	02,04,05,06,15,24, 26,28,31,32,35
Sindbad	Cinemaware	88-99	02,04,05,07,13,15,24, 26,28,31,32,35
Sky Fighter	Rushware	a.A.	28
Skyfox	Electronic Arts	71-87	04,05,06,07,10,15,18, 24,32,35
Space Battle	Kingsoft	30	19
Space Fight	Rushware	a.A.	28
Space Invasion	Micro Illus.	49	31
Star Fleet 1	Electronic Arts	118-179	05,10,15,24,32
Streetgames	Epyx	a.A.	32
Super Cycle	U.S.Gold	75	35
Starglider	Rainbird	75-80	02,03,04,06,10,15, 24,31,32,35,37
Strip Poker	Artworx	30-119	02,06,10,19,24,28, 31,32,37
Super Huey	Cosmi	69-80	06,24,31,32
Tenth Frame	Access Software	79-89	05,15,28,31,32,35,36
Thai Boxing	Kingsoft	30	19
The Three Musketiers	Comp. Novels	75	35
Trigonometry	Access Software	118	05
Turbo	Micro Illus.	49	31
Two on Two	Activision	77-99	02,03,04,10,17,32,35, 37
Video Vegas	Baudville	79	05,06,15,31,32
Winter Games	Epyx	68-129	02,04,05,06,07,15,17, 18,24,28,31,32,35,37
World Games	Epyx	59-99	02,04,05,06,07,15,17, 24,26,28,32,35,37
Adventures			
A Mind forever Voyaging	Infocom	76-129	03,04,05,06,10,15,17, 26,28,31,32,35,37
Adventure Construction Set	Electronic Arts	62-115	04,05,06,10,15,17,18, 24,28,31,32,35,37
Arazok's Tomb	Aegis	97	31
Ballyhoo	Infocom	79-88	05,06,15,18,31,32
Bard's Tale	Electronic Arts	91-128	04,05,06,10,15,18,24, 26,31,32,35,37
Black Cauldron	Sierra Online	76	04,32
Borrowed Time	Activision	62-99	01,02,03,04,05,06,10, 15,17,18,24,26,31,32, 35,37
Brataccas	Psygnosis	69-110	03,04,06,10,17,18,24, 31,35,37
Bureaucracy	Infocom	90-100	03,24,31,37
Conveted Mirror	Penguin	59	32
Crimson Crown	Penguin	59-72	15,18,32
Cutthroats	Infocom	79-109	05,06,10,15,18,32
Deadline	Infocom	79-109	05,06,10,15,18,32
Deja Vu	Mindscape	88-129	04,05,06,15,17,24, 26,28,31,32,35,37
Enchanter	Infocom	79-119	05,06,10,15,18,32
Faery Tale	Micro Illusion	90-119	04,05,07,15,24,28,31, 32,37
Hitchhikers Guide	Infocom	75-109	04,05,06,10,15,18,31, 32
Hollywood Hijinx	Infocom	88-99	02,03,05,10,15,24,31, 32,37
Infidel	Infocom	79-98	05,15,32
Jewels of Darkness	Rainbird	49-99	02,03,04,10,17,31,35, 37
Kings Quest 1	Sierra Online	95-149	04,06,10,24,26,31,32
Kings Quest 2	Sierra Online	95-99	04,06,24,31,32
Kings Quest 3	Sierra Online	95-149	04,06,10,24,31,32
Knight Orc	Firebird	87	31
Land of Legends	Micro Illus.	97	31
Leather Godesses	Infocom	75-90	03,05,06,10,15,24,31, 32,35
Mindshadow	Activision	62-99	01,02,03,04,05,06,10, 15,17,18,24,26,31,32, 35,37
Moonmist	Infocom	75-99	02,03,05,06,10,15,31, 32,35,37
Oo-Topos	Penguin	59	32
Planetfall	Infocom	75-109	05,06,10,15,18,32
Rogue	Epyx	69-99	06,10,18,24,26,28, 32,35
Seastalker	Infocom	75-88	05,06,15,18,32
Seven Cities of Gold	Electronic Arts	75-129	02,04,06,10,17,18,24, 26,28,32,35,37
Silicon Dreams	British Telecom	65-89	02,03,10,17,24,35,37
Sorcerer	Infocom	79-145	02,06,10,17,18,32,37
Space Quest	Sierra Online	119-148	06,07,24,26,32,36
Spellbreaker	Infocom	93-109	05,06,10,15,18,32
Starcross	Infocom	99-108	05,15,32
Suspect	Infocom	79-98	05,06,15,18,32
Suspended	Infocom	93-119	05,06,10,15,18,32

Titel	Hersteller	ca. Preis	Anbieter
Tass Times in Tonetown	Activision	68-99	01,02,03,04,05,06,10, 17,24,26,28,32,35,37
Temple of Apshai	Epyx	62-99	04,05,06,10,15,18,24, 26,32,35,37
The Guild of Thieves	Rainbird	89	15,31,32,37
The Pawn	Rainbird	62-106	02,03,04,05,06,10,15, 17,18,26,28,31,32,35, 37
The Quest	Penguin	59	32
Transsylvania	Penguin	59-83	06,15,18,32
Trinity	Infocom	75-100	03,04,05,06,15,24, 31,32
Ultima III Exodus	Origin Systems	119-139	05,06,10,15,18,24,28, 31,32,36,37
Uninvited	Mindscape	88-110	04,05,07,15,24,31,32, 37
Winnie the Pooh	Sierra Online	69	06
Wishbringer	Infocom	75-88	05,06,15,18,31,32
Witness	Infocom	75-145	02,05,06,15,17,18,32, 37
Zork I	Infocom	75-98	04,05,06,15,18,32
Zork II	Infocom	79-109	05,06,10,18,32
Zork III	Infocom	79-109	05,06,10,18,32
Lernprogramme			
Across the Plains		139	18
Aesop's Fables	Unicorn	97	31
Algebra	True Corp.	105-118	05,06,15,25,31,32
Algebra 2	True Corp.	105-135	06,31,32
American History Adv.	Intellectual S.	145	18
Analogies I	Intellectual S.	155	18
Analogies II	Intellectual S.	155	18
Anatonyms		85	18
Animal Kingdom	Unicorn Soft.	90	31,32
Basic Grammar		69	18
Beginning Reading I		68	18
Beginning Reading II		68	18
Beginning Reading III		68	18
Beginning Reading IV		68	18
Calculus	True Corp.	109-135	05,06,25,31,32
Capitalization		75	18
College A. Reading		159	18
Comprehensive Grammar 1	Intellectual S.	132	18
Comprehensive Grammar 2	Intellectual S.	132	18
Conversation with Comp.	Jenday Software	$29	
Decimal Dungeon	Unicorn Soft.	89-97	31,32
Discovery Math	Micro Illusions	75-79	06,18,31,32
Discovery Spell	Micro Illusions	75-79	06,18,31,32
Discovery Trivia	Micro Illusions	78	31
Discrete Mathematics	True Corp.	109-135	05,06,15,25,31,32
First Letter & Words	First Byte	97	31
First Shapes	First Byte	99	31,32
For Trade of the Great Lakes		185	18
Fraction Action	Unicorn Soft.	89-97	31,32
French Grammar I	Intellectual S.	85	18
Great States USA	The Other Guys	85-89	18,31
How a bill becomes law	Intellectual S.	$59	
Introducing Maps		145	18
Kidtalk	First Byte	109	06,31,32
Kinderama	Unicorn Soft.	89	31,32
Lessons in Reading	Intellectual S.	349	18
Lewis & Clark Expedition		215	18
Match-It	The Other Guys	78	31
Mathmagician	The Other Guys	78	31
Math Talk Fractions	First Byte	97	31
Math Talk	First Byte	97	31,32
Math Wizard	Unicorn Soft.	99	31,32
Practical Composition I	Intellectual S.	109	18
Practical Composition II	Intellectual S.	135	18
Practical Composition III	Intellectual S.	109	18
Practical Composition IV	Intellectual S.	109	18
Practical Composition V	Intellectual S.	339	18
Practical Vocabulary	Intellectual S.	132	18
Probability Theory	True Corp.	109-135	05,06,25,31,32
Punctuation		69	18
Read & Rhyme	Unicorn Soft.	79	31,32
Reading Adventure I		99	18
Reading Adventure II	Intellectual S.	145	18
Reading Adventure III	Intellectual S.	145	18
Reading and Thinking I		129	18
Reading and Thinking II		129	18
Reading and Thinking III		129	18
Senior Tutor		165	06,18
Sentence Completation		155	18
Social Studies Vocabulary		69	18
Spanish Grammar I	Intellectual S.	85	18
Spanish Grammar II	Intellectual S.	85	18
Spanish Grammar III	Intellectual S.	85	18
Speller Bee	First Byte	109	06,31,32
Spelling Detective Game		95	18
Spelling Level 2		68	18
Spelling Level 3		68	18
Spelling Level 4		68	18
Spelling Level 5		68	18
Spelling Level 6		68	18
Trancontinental Railroad		95	18
Trigonometry	True Inc.	97	25,31
Typing Tutor/Word Invader	Academy Soft.	75	18,32
U.S. Geography Adventure	Intellectual S.	145	18
Vocabulary Adventure I		145	18
Vocabulary Adventure II		145	18
Vocabulary Adventure III		145	18
Vocabulary Series (High School)		108	18
World Geography Adventure I	Intellectual S.	145	18
World Geography Adventure II	Intellectual S.	145	18
World Geography Adventure III	Intellectual S.	145	18
World Geography Adventure IV	Intellectual S.	145	18
World History Adventure	Intellectual S.	145	18
Verschiedenes			
Dr. Xes	Finally Soft.	98-119	05,15,18,31
Expert System Kit	Interactive An.	135	31
Galileo	Infinity	219	06,31,37
Krypto-Star	Krypto-Soft	198	20
LexCheck	Complete Data A.	83	31
Link Word Dutch	Artworx	58	31
Link Word French	Artworx	58	31
Link Word German	Artworx	58	31
Link Word Greek	Artworx	58	31
Link Word Italian	Artworx	58	31
Link Word Portuguese	Artworx	58	31
Link Word Russian	Artworx	58	31
Link Word Spanish	Artworx	58	31
Mad Lips	First Byte	39	31
Nancy	Finally	290	31
Perfect Score	Mindscape	155	31
Racter	Mindscape	79-148	05,06,10,15,18,26,28, 31,32,36,37
Surgeon	ISM Inc.	98-149	04,05,06,13,15,16,24, 31,37
Wow	Oxxi Inc.	76	31

01 Activision, Karlstr. 26, 2000 Hamburg 76, Tel. 040/2201370
02 Alphatron, Luitpoldstr. 22, 8520 Erlangen, Tel. 09131/25018
03 Ariolasoft, Karl-Bertelsmann-Str. 161, 4830 Gütersloh, Tel. 05241/805393
04 ASH Computerware, Von-Brandl-Str. 15, 8229 Laufen/Salzach, Tel. 08682/1327
05 Atlantis, Ernst-Reuter-Str. 151, 5030 Hürth, Tel. 02233/31066
06 B. Soltau Electronics KG, Esplanade 39, 2000 Hamburg 36, Tel. 040/340445
07 C.A.S., Sprendlinger Landstr. 71, 6050 Offenbach am Main, Tel. 069/842013
08 CompuStore, Fritz-Reuter-Str. 6, 6000 Frankfurt/Main 1, Tel. 069/567399
09 Computersysteme M. Lamm, Schönbornring 14, 6078 Neu-Isenburg 2, Tel. 06102/52535
10 Computervertrieb Krusche, Simpertstr. 3, 8110 Murnau, Tel. 08841/4276
11 CompWare H. Adler, Schlägel & Eisenstr. 9, 4352 Herten-Langenbochum, Tel. 02366/55891
12 Data Becker, Merowingerstr. 30, 4000 Düsseldorf 1, Tel. 0211/3101030
13 DTM Werbung & EDV GmbH, Bornhofenweg 5, 6200 Wiesbaden, Tel. 06121/507989
14 Grafox, Bayerstr. 57-59, 8000 München 2, Tel. 089/5306670
15 Instant Replay, Pliszstr. 38a, 4130 Moers 3, Tel. 02841/73833
16 Intelligent Memory, Basaltstr. 58, 6000 Frankfurt 90, Tel. 069/708525
17 Interplan, Nymphenburger Str. 134, 8000 München 19, Tel. 089/1234066
18 ITC, Kornstr. 34, 4154 Tönisvorst 1, Tel. 02151/803031
19 Kingsoft, Schnackebusch 4, 5106 Roetgen, Tel. 02408/5119
20 Krypto-Soft, Weizenfeld 36, 5060 Bergisch-Gladbach 2, Tel. 02202/30602
21 Markt&Technik Verlag, 8013 Haar, Hans-Pinsel-Str. 2, Tel. 089/4613-0
22 Firma Melzer, Postfach 410727, 1000 Berlin 41
23 MEV Midi-Soft, Karl-Hromadnik-Str. 3, 8000 München 60, Tel. 089/835031
24 PDC, Louisenstr. 115, 6380 Bad Homburg, Tel. 06172/24748
25 Firma J. Pfotenhauer, Postfach 1267, 7590 Achern, Tel. 07841/5056
26 Philgerma GmbH, Ungererstr. 42, 8000 München 40, Tel. 089/395551
27 Raab Bürotechnik, Friedhofstr. 36, 8605 Hallstadt, Tel. 0951/71848
28 Rushware, Daimlerstr. 11, 4044 Kaarst, Tel. 02101/60040
29 SAS-Bernd, Langgasse 93, 5216 Niederkassel 5, Tel. 0228/452626
30 Softline P. Blumenschein, Schwarzwaldstr. 8a, 7602 Oberkirch, Tel. 07802/3707
31 Softwareland, Franklinstr. 27, CH-8050 Zürich, Tel. 0041-13115959
32 Soyka Datentechnik, Hattingerstr. 685, 4630 Bochum 5, Tel. 0234/411913
33 SWS Computersysteme, Rachelweg 10, 8395 Hauzenberg, Tel. 08586/2174
34 UBM Drecker GmbH, Baaken 4, 2371 Hamdorf, Tel. 04332/1634
35 Video Loft Film, Fiedlerstr. 22-32, 3500 Kassel, Tel. 0561/877928
36 Medien-Center, Wermingser Str. 45, 5860 Iserlohn, Tel. 02371/24599
37 Jumbo-Soft, Horemansstr. 2, 8000 München 19, Tel. 089/1234065

Die Public Domain-Seite

Freie Software (oder auch Public Domain), die zum Selbstkostenpreis angeboten wird, gewinnt immer mehr an Beliebtheit. Hier erfahren Sie, warum diese Disketten so begehrt sind.

Freie Software — von Computerfreaks gerne als »Public Domain-Software« bezeichnet — sind Programme, bei denen der Programmierer auf ein Copyright verzichtet hat. Diese Software darf also frei kopiert werden. Im Laufe der Zeit erklären sich immer mehr Programmierer bereit, ihre Programme der Allgemeinheit zur Verfügung zu stellen.

Derzeit gibt es für den Amiga etwa 200 bis 250 Freesoft-Disketten, die zum Selbstkostenpreis (etwa 8 bis 12 Mark je Diskette) von mehreren Anbietern vertrieben werden.

Wir werden Ihnen auf dieser Seite einige lohnenswerte Freesoft-Disketten aus dem großen Angebot vorstellen, von denen wir meinen, daß sie einen Kauf wert sind. Zuerst stellen wir einige Disketten der Panorama-Reihe mit ihren Besonderheiten vor. (Panorama = PAcific NORthwest AMigA Users Group.)

Es handelt sich hierbei um einen Club kanadischer Computerfreaks, die den Amiga seit seinem Erscheinen beständig auf seinem Weg begleiten. So wie der Weg des Amiga mit zunächst nur spärlichen Informationen gepflastert war, so begann auch der Club mit kleinen, einfachen Programmen. Dies änderte sich jedoch bald, da der Erfahrungsaustausch mit anderen Usern und auch untereinander Früchte trug.

Der Club trifft sich regelmäßig zweiwöchentlich. Das Ergebnis dieser Treffen wird dann in entsprechenden Clubdisketten, von denen es inzwischen 36 Stück gibt, verewigt.

Auszugsweise stellen wir an dieser Stelle einige Disketten der Panorama-Serie näher vor. Diese Disketten sind deshalb interessant, da der größte Teil der auf ihnen enthaltenen Programme sinnvolle Aufgaben erfüllt und durchaus teilweise einen Platz auf der Workbench-Diskette verdient hätte. Vorteilhafterweise sind auf den neueren Disketten der Panorama-Serie zu fast allen Programmen und Routinen Textfiles enthalten, die nähere Auskunft über die einzelnen Programme geben.

Da Basic eine »Grundsprache« ist und auch mit dem Amiga ausgeliefert wird, ist Panorama »14a« ein gefundenes Fressen für jeden Basic-Programmierer. Diese Diskette enthält ausschließlich Basic-Programme. Seien es Spiele, Anwenderprogramme oder Hilfsroutinen. Empfehlenswert für jeden, der sich mit Amiga-Basic beschäftigt.

Besonderes bietet Panorama »14b«. Die Diskette enthält die Demo-Version V3.0 des Mandelbrot-Programms »MandFXP« (mit IFF-Demo-Bildern), das durch seine Schnelligkeit und seine Möglichkeiten auffällt. Leider sind in der Demo-Version wichtige Routinen wie etwa SAVE noch nicht integriert. MandFXP lohnt sich aber auf jeden Fall!

Basic-Interessierte erhalten mit dieser Diskette diverse Basic-Programme wie beispielsweise »StoneAge« oder »ScreenPrint«.

Panorama »15a« offeriert einiges. Außer 15 neuen, zum Großteil sehr schönen IFF-Bildern und einer Routine, die die Workbench modifiziert, enthält die Diskette ein Brettspiel mit sehr schöner Grafik: Cluedo. Dies ist ein Detektivspiel, bei dem es darauf ankommt, einen Mörder zu entdecken. Prädikat: Empfehlenswert.

Die Panorama-Diskette »15b« ist ein Leckerbissen für Grafikfreunde. Sie enthält einen Fractal-Generator, der zufallsgesteuert immer wieder neue Fractals erzeugt, die fast aussehen wie nach Farben aufgelöste Wärmediagramme oder Satellitenaufnahmen der Erdoberfläche. Außerdem findet man auf ihr diverse Quellcodes in »C« und neue CLI-Befehle wie RUNBACK zum Starten eines Programms vom CLI aus. Dabei kann aber nach dem Start des anderen Programms das CLI wieder geschlossen werden (normalerweise nicht möglich).

Der Shareware-Texteditor, der sich mit auf dieser Diskette befindet, gehört nicht zu den schlechtesten seiner Art, lohnt also auch den Erwerb der Panorama 15b.

Eine weitere Serie von Public Domain-Disketten ist die Serie von Fred Fish (Fish-Disks), die zum Zeitpunkt des Redaktionsschlusses 74 Disketten umfaßte. Auch diese Disketten enthalten interessante Programme, die sich nicht vor kommerzieller Software zu verstecken brauchen.

Das Public Domain-Spiel »Hack«, eine Art Adventure oder Rollenspiel, kann mit Fish »7« erworben werden. Dieses Spiel, das zwar nicht die beste Grafik besitzt, macht dafür aber unheimlich Spaß. Unsere Meinung: Sollte man nicht versäumen! Wenn jemand am C-Quellcode zu »Hack« interessiert ist, sei ihm Fish »8« ans Herz gelegt. Diese Diskette enthält den kompletten Quellcode des »Adventures«.

Mit dem »New Tek Digi-View«-Videodigitizer aufgenommene H.A.M.-Bilder enthält Fish »17«. Es ist zwar nur eine Kopie der Demo-Diskette zum Videodigitizer, doch sind die gespeicherten Bilder absolut sehenswert. Grafik-Fans sollten hier zuschlagen.

Fish »30« ist eine Sammlung von Shareware-Programmen. (Bei Shareware bitten die Programmierer, bei Gefallen einen Geldbetrag an sie zu schikken.) Diese Diskette enthält einige sehr nützliche Programme:

»StarTerm V3.0« ist ein Terminalprogramm, mit dem sich eine Menge anfangen läßt. Tip: Lohnt sich auf jeden Fall.

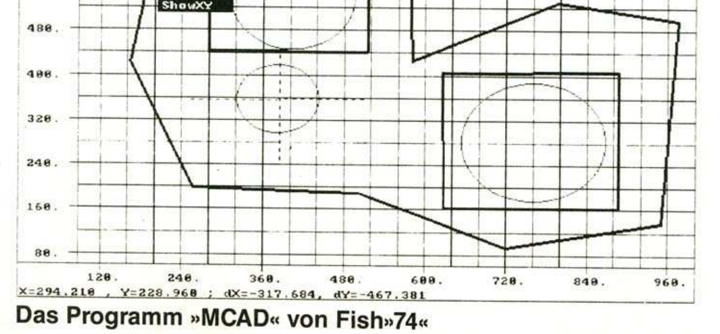
Der ebenfalls auf dieser Diskette zu findende »MenüEditor« generiert Menüs, die anschließend als C-Quellcode gespeichert werden. Gutes, wenn auch noch nicht ganz vollendetes Werkzeug für Programmierer.

Erwähnenswert ist auch noch der »FontEditor«. Dieses Programm ist ein verbesserter Zeichensatz-Generator und sollte auch in keiner Sammlung fehlen.

Unsere Meinung: Nicht lebensnotwendig, aber gut.

Eine Überraschung hatte Fish »74« für uns parat: Auf dieser Diskette befindet sich ein CAD-Programm, welches es durchaus mit kommerziellen Zeichenprogrammen aufnehmen kann (siehe Bild). Das Programm erinnert uns stark an »Aegis Draw«. Es macht einen absolut professionellen Eindruck und läßt kaum Wünsche für ein Programm dieser »Preisklasse« offen.

Wir hoffen, daß Sie durch diese Seite etwas auf die Public Domain-Szene aufmerksam geworden sind. Sollten Sie Informationen über neueste oder exotische Freesoft besitzen, schreiben Sie uns doch bitte. Nur so können wir die anderen Leser, die vielleicht mehr wissen möchten, gut genug informieren. (dm)



Das Programm »MCAD« von Fish»74«

Anbieter von Public Domain-Disketten (ohne Anspruch auf Vollständigkeit):

Atlantis, Ernst-Reuter-Str. 151, 5030 Hürth, Tel. 02233/805393

Christian Bellingrath, Trift 10, 5860 Iserlohn, Tel. 02371/24192

EcoSoft AG, Kaiserstr. 21, 7890 Waldshut, Tel. 07751/7920

Fastworks, Fichtestr. 16, 5090 Leverkusen 1, Tel. 0214/92802

Stefan Ossowski, Veronikastr. 33, 4300 Essen, Tel. 0201/788778

Uwe Schmielewski, Haroldstr. 71, 4100 Duisburg 1, Tel. 0203/376448

Soyka Datentechnik, Hattinger Str. 685, 4630 Bochum 5, Tel. 0234/411913

AB Computersysteme, A. Büdenbender, Wildenburgstr. 21, 5000 Köln 41, Tel. 0221/4301442

C.A.S., Sprendlinger Landstr. 71, 6050 Offenbach, Tel. 069/842013

Jumbo Soft, Horemansstr. 2, 8000 Müchen 19, Tel. 089/1234065

AMIGA SOFTWARE ZUM ABHEBEN

SPIELE

Titel	Preis
A Mind Forever Voyaging	DM 89,00
Adventure Constr. Set	DM 79,00
Alien Fires	DM 89,00
Archon I	DM 79,00
Archon II	DM 79,00
Arcticfox	DM 69,00
Arena	DM 79,00
Balance of Power	DM 89,00
Bard's Tale	DM 97,00
Borrowed Time	DM 79,00
Brataccas	DM 79,00
Bureaucrazy	DM 96,00
Ch.Basketb. Two on Two	DM 79,00
Championship Baseball	DM 79,00
Championship Football	DM 85,00
Championship Golf	DM 79,00
Chessmaster 2000	DM 88,50
Deep Space	DM 88,50
Defenders of the Crown	DM 87,50
Déjà Vu	DM 88,50
Fairy Tale	DM 98,00
Flightsimulator II	DM 118,00
Scenerys Disk Flightsim.II	DM 59,00
Flip Flop	DM 29,95
Galileo	DM 215,00
Grand Slam Tennis	DM 79,00
Guild of Thieves	DM 89,00
Hacker I	DM 69,00
Hacker II	DM 69,00
Halley Project	DM 69,00
Hollywood Hijinx	DM 89,00
Hollywood Poker	DM 59,00
Jewels of Darkness	DM 48,00
Karate King	DM 48,00
King of Chicago	DM 99,00
Leaderboard Golf	DM 74,90
Leaderboard Tournament	DM 47,00
Little Computer People	DM 89,00
Marble Madness	DM 68,00
Mind Breaker	DM 29,95
Mindshadow	DM 69,00
Moonmist	DM 86,00
One on One	DM 98,00
Pac Boy	DM 29,95
Planetfall	DM 99,00
Portal	DM 89,00
Programm des Lebens	DM 155,00
Quintette	DM 99,00
Quiwi	DM 48,00
Racter	DM 99,00
Rocket Attack	DM 29,95
Seven Cities of Gold	DM 89,00
Shanghai	DM 79,00
Silicon Dreams	DM 69,00
Sinbad and t. t. of t. falcon	DM 87,00
Sky Fighter	DM 59,00
Sorcerer	DM 99,00
Space Fighter	DM 29,95
Starglider	DM 79,00
Strip-Poker	DM 79,00
Surgeon	DM 149,00
Tass Times In Tonetown	DM 75,00
Temple of Apshai Trilogy	DM 78,00
The Pawn	DM 75,00
The Three Musketeers	DM 74,00
The Witness	DM 99,00
Ultima III	DM 99,00
Uninvited	DM 89,00
Winter Games	DM 64,00
World Games	DM 64,00

GRAFIK

Titel	Preis
Animator + Images	DM 275,00
Art Disk EA	DM 65,00
Art Disk Print I	DM 65,00
Art Pack 1 Aegis	DM 65,00
Art Parts Volume 2	DM 68,00
Butcher	DM 89,00
Calligrapher	DM 218,00



Software Verlag GmbH • 8000 München 19
Horemansstraße 3

Das Jumbo-Angebot des Monats!

Deluxe Video 1.2 DM 228,--
Video-Animation zu Erzeugung von Videovorspännen.
Sehr komfortabel! Das reinste Ministudio.

Alien Fires DM 89,--
Brillantes Rollenspiel mit super Grafik und super Sound!

Titel	Preis
Deluxe Paint II	DM 239,00
Deluxe Print	DM 139,00
Deluxe Video V1.2	DM 228,00
Digi View	DM 465,00
Draw	DM 239,00
Draw plus	DM 479,00
Dynamic CAD	DM 998,00
Impact	DM 199,00
Page Setter (deutsch)	DM 349,00
Prism	DM 159,00
Title Construction Set	DM 95,00
TV-Text 3D	DM 245,00
Zuma Fonts 1	DM 69,00
Zuma Fonts 2	DM 69,00
Zuma Fonts 3	DM 69,00

MUSIK

Titel	Preis
Deluxe Music Constr.	DM 229,00
Future Sound	DM 395,00
Instant Music	DM 79,00
Instant Music Song Disk	DM 79,00
It's just Rock'n'Roll	DM 69,00
Midi-Interface	DM 99,00
Music Studio	DM 99,00
Pro Midi Studio	DM 349,00
Sonix	DM 178,00
Sound Sampler	DM 219,00

KOMMERZIELL

Titel	Preis
Analyze V 2.0	DM 349,00
BBS Bulletin Board	DM 210,00
dBMan	DM 345,00
Flow	DM 225,00
Laserscript	DM 99,00
Logistix (deutsch)	DM 329,00

Titel	Preis
OnLine!	DM 155,00
Organize!	DM 228,00
Pro Write	DM 289,00
Publisher 1000	DM 459,00
Pagesetter	DM 349,00
Scribble 2.0	DM 219,00
Softwood File II GS	DM 229,00
Superbase	DM 245,00
UBM-Text V2.2	DM 229,00
Vip Profesional (engl.)	DM 329,00
Vizawrite (deutsch)	DM 498,00

SPRACHEN UND TOOLS

Titel	Preis
AC Basic	DM 412,00
Amiga C Compiler	DM 479,00
C-Monitor V. 2.00	DM 195,00
Cambridge Lisp	DM 479,00
CLI mate	DM 78,00
Digital Link	DM 149,00
Fortran 77	DM 649,00
Gizmoz V2.0	DM 155,00
Grabbit	DM 69,00
Lattice C Compiler V3.1	DM 438,00
Macro Assembler	DM 179,00
Manx Aztec C Com.	DM 1120,00
Manx Aztec C Devel.	DM 648,00
Manx Aztec C Personal	DM 445,00
MCC Pascal	DM 248,00
Modula-2 Commercial	DM 619,00
Modula-2 Developers	DM 349,00
Modula-2 Standard	DM 219,00
Multi Forth-83	DM 578,00
Shell CLI	DM 118,00
Toolkit	DM 99,00
True Basic Interpreter	DM 349,00
UCSD-Pascal	DM 348,00
Zing! (CLI deluxe)	DM 179,00

BÜCHER

Titel	Preis
Amiga 500 für Einsteiger	DM 39,00
Amiga Basic	DM 59,00
Amiga für Einsteiger	DM 49,00
Amiga Maschinensprache	DM 49,00
Amiga Tips und Tricks	DM 49,00
Amiga DOS-Manual	DM 79,90
Amiga DOS Express	DM 79,90
C für Einsteiger (Amiga)	DM 39,00
Das können Amiga 500 und 2000	DM 29,00
Das Amiga Handbuch	DM 49,00
Das Amiga Grafik Buch	DM 49,00
The Amiga Book	DM 69,00
Hardware Ref. Manual	DM 62,50
Intuition Ref. Manual	DM 62,50
Programmers Guide	DM 50,80
Programmers Handbook	DM 99,90
ROM-Kernel Ref.: exec	DM 62,50
& Libraries und Devices	DM 88,00

DEMO DISKETTEN

Titel	Preis
Draw	DM 12,00
Animator	DM 12,00
Gridiron	DM 12,00
Zuma Fonts	DM 12,00
TV Text	DM 12,00
Investor	DM 12,00
Adventure Const. Set	DM 12,00
Bards Tale	DM 12,00
Zing!	DM 12,00
Acquisition	DM 12,00
Dynamic CAD	DM 12,00
Flow	DM 12,00
Digi-View (H.A.M.)	DM 12,00
Perfect Sound	DM 12,00
Title Const. Set	DM 12,00
Modula	DM 12,00
dBMan	DM 12,00
Vizawrite	DM 12,00
Amilink News	DM 12,00
Pro Write	DM 12,00
Logistix	DM 12,00

Preisänderungen vorbehalten.
Lieferung nach Verfügbarkeit.

61

Name
Vorname
Straße
PLZ/Ort
Hiermit bestelle ich:

Ich zahle per:
- [ ] beiliegendem Verrechnungsscheck
- [ ] Nachnahme (zuzügl. N.N-Gebühren)

zuzügl. Versandkosten von DM 6,50

Jumbo Soft
Software Verlag GmbH
Horemansstr.3, 8000 München 19
Tel. 089/1 23 40 65

Komfort im CLI

AMIGA TEST

Vom hochgelobten Bedienungskomfort des Amiga ist im CLI nicht viel übriggeblieben. Eine »Muschel« verspricht Abhilfe.

Fast alle Compiler für den Amiga müssen über das CLI und die Tastatur gesteuert werden; viele andere Dinge sind über die Workbench nur schwer oder gar nicht machbar. Doch leider bietet das CLI als Benutzerschnittstelle zum Computer kaum Bedienungskomfort, es ist zum Beispiel nicht einmal möglich, mit dem Cursor in einer gerade geschriebenen Kommandozeile zurückzufahren, ohne das bereits Geschriebene zu löschen, wenn am Anfang der Zeile ein Fehler entdeckt wurde.

Daß es auch anders geht, beweist die vom Prinzip her ähnlich aufgebaute Benutzerschnittstelle der MS-DOS-Rechner, die wesentlich komfortabler als das CLI ist. Sie hat wohl auch die Programmierer von Metacomco dazu angeregt, mit der »Metacomco Shell« die Bedienung des CLI sehr viel einfacher und schneller zu machen.

Auf der nicht kopiergeschützten Programmdiskette befinden sich einige Dateien, die durch ein Installationsprogramm automatisch auf jede beliebige Workbench-Diskette geschrieben werden können. Dabei werden die Versionen 1.1 und 1.2 der Workbench unterstützt. Dann steht dem vom CLI frustrierten Programmierer ein Plus an Komfort zur Verfügung. Alle neuen Befehle bleiben als Teil des Shell-Programms ständig im Speicher.

Befehle wiederholen

Die Shell erlaubt es, den Cursor in einer bereits eingegebenen Zeile (vor dem Drücken von <RETURN>) hin und her zu bewegen und Zeichen zu ändern, einzufügen oder zu löschen. Hat man <RETURN> gedrückt und enthält die Zeile einen Fehler, so läßt sie sich mit Cursor-Up wieder auf den Bildschirm bringen und korrigieren.

Jede eingetippte Kommandozeile wird automatisch in einem Zwischenspeicher abgelegt. Dieser Speicher läßt sich mit einem Kurzkommando auf den Bildschirm bringen. Nun kann ein Kommando zur nochmaligen Verwendung ausgesucht werden. Dabei kann man entweder die Nummer oder die Anfangsbuchstaben des Kommandos eingeben, welches dann entweder sofort ausgeführt oder editiert und dann ausgeführt werden kann. Mit den Cursor-Up und Cursor-Down-Tasten können die gespeicherten Kommandos ebenfalls »durchgeblättert« und ausgeführt werden. Normalerweise speichert die Shell die letzten 10 eingegebenen Kommandos, diese Zahl ist aber veränderbar.

Eines der unpraktischen Dinge am CLI ist, daß jedes eingetippte Kommando erst einmal von der Diskette geladen werden muß, was Besitzer von nur einem Diskettenlaufwerk zu Diskjockeys macht und eine Menge Zeit verschwendet. Hier schafft die Shell mit dem »resident«-Kommando Abhilfe. Dieses Kommando kann auf die meisten Systembefehle und auf viele andere Programme angewandt werden. Es lädt diese in den Speicher, wo sie resident bleiben. Der Vorteil gegenüber einer RAM-Disk liegt darin, daß ein Programm, wenn es ausgeführt wird, nur einmal im Speicher steht und nicht doppelt wie bei einer RAM-Disk. Außerdem fällt die Ladezeit von der RAM-Disk weg. Ein residentes Programm kann sogar von mehreren Tasks aus gleichzeitig verwendet werden, ohne mehr Speicherplatz als unbedingt nötig zu brauchen. Wird ein residentes Programm nicht mehr benötigt, kann mit einem weiteren Kommando sein Speicherplatz wieder freigegeben werden.

Residente Befehle

Um das Eintippen von Kommandozeilen abzukürzen, gibt es verschiedene Möglichkeiten. Einerseits kann man oft verwendeten Kommandos einen Kurzcode zuweisen, was die Tipparbeit schon etwas erleichtert. Dieses Verfahren ist aber nur auf Kommandonamen anwendbar, die natürlich keine Leerzeichen enthalten dürfen. Andererseits können Variablen definiert werden, die alle verfügbaren Zeichen enthalten dürfen und hauptsächlich für Filenamen und ähnliches brauchbar sind. Mit dem Befehl »set s df1:src/window.asm« wird zum Beispiel eine Variable »$s« definiert, die dann in allen Kommandos statt dem Ausdruck »df1:src/window.asm« gebraucht werden kann. Da sich theoretisch beliebig viele Variablen und Kommandoabkürzungen definieren lassen, steht einer Verminderung des Tippaufwandes nichts mehr im Weg. Es ist auch möglich, bereits definierte Variablen bei der Definition einer neuen Variablen einzusetzen, was zu einem interessanten Effekt führt: Da die neu definierte Variable die Ursprungsvariable enthält, verändert sich bei einer Änderung der ursprünglichen Variable auch die neu definierte. Somit wirkt sich etwa die Änderung einer Variable, die einen Filenamen enthält, auf alle anderen Variablen aus, die die Variable mit dem Filenamen benutzen. Das klingt recht kompliziert, wird aber nach kurzer Einarbeitung zur praktischen Hilfe bei der täglichen Arbeit.

Eine andere Möglichkeit, sich Tipparbeit zu sparen, sind die nun freiprogrammierbaren Funktionstasten. Die Kommandos dürfen auch Leerzeichen enthalten und können mit einem automatischen Return abgeschlossen werden.

All diese Möglichkeiten wären aber relativ sinnlos, wenn nach dem Neustart des Systems alle Variablen, Kommandoabkürzungen und die Funktionstastenbelegung wieder neu definiert werden müßten. Deshalb gibt es die Möglichkeit, beim Start der Shell automatisch ein Batchfile ablaufen zu lassen, in dem alle Einstellungen vorgenommen werden. Dieses System arbeitet wie die »Startup-Sequence«, wobei das Einstellungsprogramm für die Shell aber an keinen bestimmten Namen gebunden ist. Hat man sich nun in mühevoller Arbeit die richtigen Variablen und so weiter etwa zum Arbeiten mit einem Assembler aufgebaut, so können Listen der Variablen, Kommandoabkürzungen und Funktionstasten auf der Diskette abgelegt werden.

Funktionstasten belegen

Wie jeder erfahrene CLI-Benutzer weiß, wird ein auszuführendes Kommando, das von der Diskette geladen werden muß, zuerst im momentan aktiven Directory und dann im C-Directory gesucht. Manchmal ist es aber sinnvoll, in diesen Suchpfad auch noch andere Directories einzubauen. Zu diesem Zweck bietet die Shell den »PATH«-Befehl an, der den Suchpfad um beliebig viele Directories erweitert.

Zum Arbeiten mit Subdirectories dient auch der bekannte »CD«-Befehl. Um den Umgang mit den häufig genutzten Subdirectories zu erleichtern, wird dieser Befehl bereits beim Laden der Shell automatisch resident gemacht. Oft ist es aber nötig, aus einem Subdirectory in ein anderes und dann wieder zurück zu springen, was bei längeren Directorynamen etwas umständlich werden kann. Dafür wurde der CD-Befehl durch den PUSH-Befehl ergänzt, der zuerst einmal einen Sprung in ein anderes Directory durchführt. Das vorherige Directory wird gespeichert und kann mit dem POP-Befehl ohne Parameter schnell wieder erreicht werden. Zusätzlich wird das Arbeiten durch die ständige Anzeige des aktuellen Unterverzeichnisses in der Titelzeile wesentlich erleichtert.

Die Anleitung faßt die vorhandenen Befehle auf 24 Seiten in englischer Sprache relativ knapp zusammen. Die Metacomco Shell selbst macht diese Knappheit aber durch sehr weitgehende Möglichkeiten wieder wett, der ermüdenden Tipperei im CLI zu entsagen und sich statt dessen auf das Wesentliche zu konzentrieren. (Andreas Lietz/rb)

Anbieter (Adressen siehe Marktübersicht): Atlantis, B. Soltan Electronics, DTM, Philgerma GmbH, Softwareland, Soyka Datentechnik, SWS Computersysteme, Jumbo-Soft
Preis: 118 bis 169 Mark

Dr. Hacker ... zum OP bitte!

AMIGA TEST

»The Surgeon« soll die Simulation einer chirurgischen Operation ermöglichen. Hält Surgeon, was die Werbung verspricht?

Schon wieder ein Spiel, werden Sie jetzt vielleicht denken. Aber dem ist nicht so. »The Surgeon« ist nicht nur ein Spiel. Es ist gleichzeitig eine komplette Simulation einer Operation am Menschen. Man lernt Organe und Arzneien kennen und ihre Funktionen verstehen. Sie werden selbst zum gefürchteten »Menschenmetzger« und beginnen (wenn auch nur in sehr begrenztem Umfang) zu verstehen, mit welchen Problemen sich ein Chirurg herumschlagen muß. Und zu guter Letzt macht es einfach Spaß, einmal in die Welt der sonst so ernsten Chirurgie auf ganz unverfängliche Art hineinzuschnuppern. Doch nun zum Programm selbst: Nach dem Laden wird als eine Art Kopierschutz ein bestimmtes Wort aus der Anleitung abgefragt. Hat man diese etwas unpraktische Hürde überwunden, begrüßt einen der Computer mit einer angenehm klingenden, digitalisierten Stimme. Man sollte auch immer darauf achten, daß genügend Speicher vorhanden ist, sonst werden manche Bilder nicht korrekt dargestellt, was leider nicht durch irgendeine computerseitige Meldung oder einen Vermerk in der Anleitung erwähnt wird. Als erstes wird man mit dem zu behandelnden Fall vertraut gemacht. Dazu erhält der Anwender ein Röntgenbild des Beckenbereiches und die Aussage des Patienten (Bild 1). Nun kann man entscheiden, ob dem Patienten eine Medizin verordnet werden soll, oder ob man gleich operieren muß. Außerdem kann der Patient unter Beobachtung gehalten werden und macht dann später weitere Aussagen über seinen Zustand. Sollte man sich zur Operation entschließen, wird es kritisch. Spätestens hier sollte nämlich die 20seitige Anleitung zur Hand genommen werden — und ein großes Wörterbuch am besten gleich dazu. Die Anleitung ist nur in englisch geschrieben und selbst jemand mit guten Englischkenntnissen hat sicherlich Probleme beim Lesen und Verstehen der medizinischen Fachausdrücke, die in normalen Wörterbüchern nicht vorkommen. »The Surgeon« ist sicherlich zur Zeit eines der Programme, die eine deutsche Übersetzung der Anleitung am nötigsten haben. Aber auch hier bringt eifriger Fleiß bald den gewünschten Erfolg und man setzt zum ersten Mal das Skalpell an. Aber Achtung, vorher Hände und Wunde desinfizieren und unbeteiligte Körperflächen abdecken. Eventuell sollte man dem Patienten vorher noch Antibiotika verabreichen (je nach Zustand).

Jetzt kann endlich das Skalpell angesetzt und die Bauchdecke geöffnet werden. Sofortige Säuberung der Wunde von ausströmenden Blut ist ein absolutes Muß. Und daß die Blutung möglichst bald gestoppt werden sollte, ist wohl auch selbstverständlich. Dann Haut und Fettschicht entfernen und die Blutung nochmals kontrollieren. Die Wunde wird nun mit Hilfe von Klammern aufgehalten. Jetzt das Bauchfell aufschneiden und festklammern.

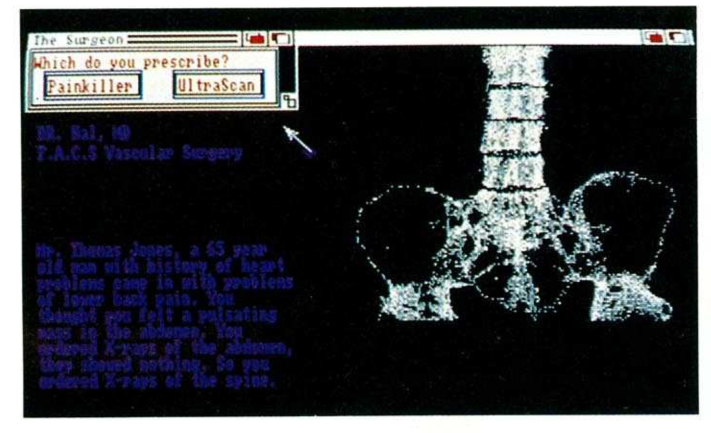


Bild 1. Die Diagnose wird mit Hilfe des Röntgenbildes gestellt

Nur mit Fingerspitzengefühl ...

Da dieser Vorgang recht großen Druck benötigt, kann es hier passieren, daß der Dünndarm mit angeschnitten wird. Dieser muß (wegen der eventuell dort vorhandenen Speisereste und einer daraus resultierenden Infektionsgefahr) sofort wieder geschlossen und genäht werden. Während der ganzen Operation, bei der man verständlicherweise rasch und zügig arbeiten muß, sollte man ständig das EKG (Elektrokardiogramm), welches in Form eines Oszilloskopes die Herztätigkeit des Patienten darstellt, im Auge behalten. Außerdem ist die Überwachung des Blutdruckes und Maßnahmen zu dessen Ausgleich ratsam.

Nun wird es richtig ernst: Dünndarm zur Seite schieben, Zwölf-Fingerdarm aus dem Weg räumen und Heparin, das die Verklumpung des Blutes unterdrückt, spritzen. Danach kommt es fast zwangsweise zu einer Erweiterung der Aorta, was natürlich sofort behandelt werden muß. Dazu müssen die Arterien, die in das rechte und linke Bein weiterführen, kurzzeitig abgedrückt werden (mit speziellen Klammern). Des weiteren muß die Blutzufuhr oberhalb der Verdickung unterbrochen werden, da der Patient ansonsten sofort verbluten würde. Nun kann die Blase, die sich durch die Erweiterung der Aorta und der nachfolgenden Bildung eines Blutklumpens gebildet hat, geöffnet werden. Dazu wird mit der Schere ein kleiner Schnitt gemacht, der gerade groß genug ist, um den Blutklumpen mit der Hand zu entfernen. Dann geht es ans Ziel dieser ganzen Operation: Den geschädigten Teil der Aorta ganz entnehmen und das Transplantat einsetzen. Oberen Teil mit fünf Stichen annähen und die in die Beine führenden Arterien in der selben Weise befestigen. Zum Ende der Operation werden nach und nach die Klammern entfernt, die einzelnen Lagen Fett und Bauchfell wieder geschlossen und vernäht (dabei unbedingt darauf achten, keine medizinischen Geräte zurückzulassen). Als letztes wird die Haut wieder geschlossen und mit einigen speziellen Klammern vernäht. Hat der Patient bis hierher überlebt — Herzlichen Glückwunsch. Stirbt er aber während der Operation, ertönt ein markerschütternder Schrei aus den Tiefen des Amiga.

Der Nachteil des Programmes wird aus der obigen Beschreibung einer Operation ersichtlich: es gibt nur einen Patienten und nur eine Krankheit, an der dieser leidet. Sollte man einige Operationen erfolgreich durchgeführt haben, verliert »The Surgeon« irgendwann seinen Reiz.

Die Grafiken sind äußerst genau gezeichnet und durchaus des Amiga »würdig«. Der Sound kommt, verständlicherweise, bei diesem Programm etwas zu kurz. Aber was hört man im OP schon anderes als das (hoffentlich immer) monotone Signal des EKG. Ein Nachteil, der sich allerdings leicht beheben lassen sollte, liegt aber in der englischen Anleitung. Vielleicht lassen sich die Verantwortlichen ja durch diesen Bericht dazu verleiten, jedem Programm eine deutsche Übersetzung beizulegen. Denn die Anleitung ist wirklich nicht schlecht und alle Begriffe werden ausführlich erläutert, so daß man auch beim Studieren derselben schon recht viel über den menschlichen Körper lernt. Die recht ungewöhnliche, aber nichts desto trotz interessante Programmidee macht »The Surgeon« zu einem Programm für den Amiga, hinter dem weit mehr steckt, als bei den meisten Spielen oder Simulationen. Ein Software-Konzept, das hoffentlich bald viele Nachahmer findet.

(Ottmar Röhrig/jk)

Anbieter (Adressen siehe Marktübersicht): ASH-Computerware, Atlantis, B. Soltau Electronics, DTM, Instant Replay, Intelligent Memory, Softwareland, PDC, Jumbo Soft
Preis: 98 bis 149 Mark

Die Tastatur im Griff mit »Zing!Keys«

AMIGA TEST

Wer bisher glaubte, er habe schon alles gesehen, was die Multitasking-Fähigkeiten des Amiga betrifft, der täuscht sich gewaltig. Zing!Keys ist ein formbares Software-Werkzeug mit erstaunlichen Fähigkeiten.

Multitasking scheint ein Zauberwort bei Computern der gehobenen Klasse geworden zu sein. Ganz besonders gilt dies natürlich für den Commodore Amiga, der dies auch für den Heimbereich bereitstellt. Im wesentlichen bezeichnet Multitasking den scheinbar gleichzeitigen Ablauf mehrerer Programme in einem Computer. Völlig gleichzeitig laufen die Programme natürlich nicht ab. In Wirklichkeit kann der Computer durch einen extrem schnellen Prozessor und durch die Unterstützung von entsprechenden Coprozessoren jedes ablaufende Programm eine Weile abarbeiten um dann zum nächsten zu wechseln. Für den Benutzer sind die kurzen Verzögerungen nicht mehr wahrnehmbar. Drucker-Spooler sind eine ganz einfache Anwendung des Multitasking: Während ein Programm läuft, druckt der Computer gleichzeitig lange Listen auf einem Drucker aus. Beim Amiga dürfen sogar in verschiedenen Fenstern und Bildschirmen die unterschiedlichsten Programme ablaufen, wobei allein der Benutzer entscheidet, mit welchem Programm er sich gerade beschäftigen möchte.

Einer nach dem anderen

Bisher mußte man allerdings bei der Software, die für den Amiga geschrieben wurde, auf eine gute Ausnutzung der Multitasking-Fähigkeiten verzichten. Nicht nur bei Spielen, sondern auch bei vielen Utilities ist es nicht möglich, noch zusätzlich andere Programme zu starten. Auch wenn man über genügend Speicherplatz in Form einer RAM-Erweiterung verfügt, lassen die meisten Programme nicht zu, daß sich noch ein anderes Programm im Speicher befindet. Mit Zing! einem Werkzeug für das CLI des Amiga (Test in Amiga, Ausgabe 6/7) hat man sich schon in die richtige Richtung bewegt. Ein kompromißlos modularer Aufbau und ein System von auf Tastendruck verfügbaren Tools machen das Grundkonzept aus. Unterstützt durch das Multitasking arbeitet Zing! Keys nun auf ähnliche Weise mit einem speicherresidenten Programmteil, in dem der Benutzer nach Belieben Tasten mit Befehlen belegen kann. Dies sind natürlich hauptsächlich Kombinationen aus mehreren verschiedenen Tasten, die gedrückt werden müssen, um eine vorher bestimmte Sequenz von Befehlen aufzurufen. Im Prinzip klingt dies ziemlich simpel: Ein Programm, mit dem man eigene Funktionstasten definieren und belegen kann. Das Konzept hat sich jedoch schon bei Zing! bewährt und man kann hier jegliche denkbare Befehlskombination inklusive Programmaufruf und Parameterübergabe realisieren.

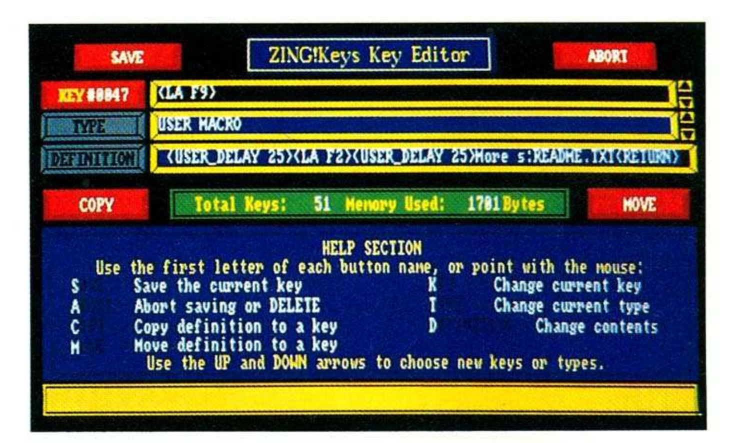


Bild 1. Das Menü des Tasten-Editors von Zing!Keys

Damit Zing!Keys mit möglichst vielen Programmen zusammenarbeitet, stehen für die Belegung aber nicht nur die zehn Funktionstasten des Amiga zur Verfügung, sondern auch die Tasten <ALT>, <CTRL>, <SHIFT>, <CAPS LOCK> und die speziellen AMIGA-Tasten. Außerdem gehören auch noch beide Mausknöpfe zu diesen Sondertasten, die alle als sogenannte »Qualifier« bezeichnet werden. Das heißt in Verbindung mit den normalen Tasten können Tastenkombinationen definiert werden. Zwischen den Tasten rechts und links auf der Amiga-Tastatur wird natürlich auch noch unterschieden. Es ist also ein Unterschied, ob man zum Beispiel die F10-Taste zusammen mit der linken oder rechten Shift-Taste drückt. Wenn man dies einmal durchrechnet, kommt dabei schon eine beachtliche Anzahl an möglichen Tastenkombinationen heraus, die dem Benutzer dann zur Verfügung stehen und die sich frei mit Befehlen jeglicher Art belegen lassen.

Wie sieht nun die praktische Anwendung von Zing!Keys aus und was läßt sich wirklich damit realisieren? Man sollte grundsätzlich nicht erwarten, bei jeglicher Art von Programmen Utilities über Tastenkombinationen im Hintergrund abrufbereit installieren zu können. Dies funktioniert wie vorab schon gesagt nur mit Programmen, die das Multitasking des Amiga auch unterstützen und nicht unterdrücken. Für eine Textverarbeitung wie zum Beispiel »Prowrite« wäre dies jedoch denkbar. Über den Zing!Keys-Editor kann man auf komfortable Weise Tastenbelegungen erstellen (Bild 1). Diese Kombinationen werden dann auf Diskette in der »Startup-Sequence« unter dem Namen »Initial.Keys« gespeichert. Zusätzlich braucht man noch das Hauptprogramm, welches auch zu Beginn geladen werden sollte und das die jeweilige Tastenbelegung dann abrufbereit im Speicher hält. Im Prinzip braucht also nur eine Zeile in die jeweilige Startup-Sequence eines Programmes eingebaut werden, die das Zing!-Keys-Programm mit aufruft. Danach hat man seine eigene zusätzliche Tastenbelegung zur Verfügung. Wer meint, daß einige Programme ja auch bestimmte festgelegte Belegungen haben und daß diese mit den Zing!Keys kollidieren könnten, der sei unbesorgt. Die selbsterstellte Tastenbelegung geht vor. Es gibt jedoch einen Befehl, der entweder für den nächsten Tastendruck die eigene Belegung aufhebt oder sogar ganz abschaltet. Dann kann man auf die ursprüngliche Verwendung der Tasten der jeweiligen Utility wieder zugreifen. Zing!Keys unterscheidet drei Kategorien von Tastenkombinationen. Zum einen sind dies die »Hot Keys«. Sie lassen den Aufruf von einzelnen ganz bestimmten Programmen zu.

Programmaufruf per Taste

Dazu gehört zum Beispiel ein kleines aber sehr nützliches Werkzeug zum Festhalten des momentan sichtbaren Bildschirmes, den man dann ausdrucken oder speichern kann. Des weiteren gibt es die »Macros«, wobei noch zwischen »User Macros« und »System Macros« unterschieden wird.

Ein System-Macro bezieht sich auf vorgefertigte Funktionen und Aufgaben, die Zing! Keys von sich aus anbietet. Diese sind in das Zing!Keys-Hauptprogramm integriert und brauchen daher nicht noch extra nachgeladen zu werden. Darunter befinden sich Funktionen wie »Expand Window«, die das momentan aktive Fenster auf volle Bildschirmgröße bringt oder »Cycle Forward«,

womit sich überlagernde Fenster austauschen lassen. Dies sind natürlich nur die einfachsten Beispiele. Man sollte nicht zu der Auffassung gelangen, daß man solche Funktionen, wie ein Fenster auf maximale Größe bringen, viel eher mit der Maus direkt am Bildschirm vollziehen kann und sollte. Der Sinn dieser Funktionen bei Benutzung von Zing!Keys liegt vielmehr darin, daß sich auch mehrere davon hintereinander auf einen Tastendruck abrufen lassen und somit vorprogrammierte Abläufe automatisch stattfinden. Beispiel: Beim Formatieren einer Diskette braucht man sich nicht mehr darum zu kümmern erst ein CLI-Fenster zu öffnen und dann die entsprechenden Befehle einzutippen. Dies ist ein vorprogrammierbarer Ablauf, den man dann mit einem einzigen Tastendruck aktivieren kann. Sogar eine beliebige Zeitverzögerung darf in den Ablauf dieser »Mikro«-Programme eingebaut werden, falls zum Beispiel bei Diskettenoperationen etwas Zeit vergehen muß, bis der nächste Befehl akzeptiert werden kann.

Die andere Form von Macros sind die »User-Macros«, bei denen eine beliebige Abfolge von eigenen Befehlen und Programmaufrufen sowie auch System-Macros erlaubt sind. User Macros dürfen sich sogar selbst aufrufen und eigene Macros definieren, was ein unglaublich variables System zur Verfügung stellt. Zing!Keys ist ein Programm, in das man sich erst einarbeiten muß. Die Befehlssyntax und die Abkürzungen, die man verwenden muß, um einen bestimmten Erfolg zu erzielen, wollen erst einmal verstanden und gelernt sein, obwohl sie natürlich einzeln im Handbuch ausführlich erklärt sind. Hat man den Einstieg gefunden, eröffnet Zing!Keys jedem Anwender, der das Multitasking des Amiga ausnutzen möchte, ungeahnte Möglichkeiten. (jk)

Info: Markt & Technik Verlag AG, Hans-Pinsel-Str. 2, 8013 Haar, Preis: 179 Mark

---

Aegis Impact

AMIGA TEST

Gerade durch den neuen Amiga 2000 wird die hervorragende Grafik der Amiga-Serie auch für berufliche Anwender interessant. Das Programm »Aegis Impact« erstellt Geschäftsgrafiken aller Art.

Aegis Impact wurde zu dem Zweck entworfen, aus langweiligem Zahlenmaterial interessante farbige Grafiken zu erstellen. Dabei steht natürlich an erster Stelle die Eingabe des zu erfassenden Zahlenmaterials. Das Programm kann bis zu acht »Serien«, das heißt gleichartige Daten, die zusammen in einer Grafik gezeigt werden sollen verwalten. Pro Serie werden bis zu zwanzig verschiedene Daten verwaltet (zum Beispiel die Verkaufszahlen mehrerer Produkte innerhalb eines Jahres), so daß sich die Verkaufszahlen von acht verschiedenen Produkten innerhalb von 20 Monaten miteinander in einer Grafik zeigen lassen. Die Daten zur Erstellung der Grafiken können entweder über die Tastatur eingegeben oder von der Diskette importiert werden. Hat man die benötigten Zahlen erst einmal eingegeben (zum Beispiel »Januar: 167 Stück«), so lassen sich aus diesem Zahlenmaterial eine Menge Grafiken erstellen.

Die eingegebenen Daten sind nun in drei verschiedenen Formaten darstellbar: als Balken-, Linien- oder Kuchengrafik. Jeder dieser Grundtypen kann nochmals variiert werden.

Bei der Balkengrafik sind die Darstellungsmöglichkeiten vielfältig: Die Balken können vertikal oder horizontal auf dem Bildschirm angeordnet werden und lassen sich durch einen einfachen Mausklick dreidimensional darstellen. Die Balkengrafik ist sicherlich eine der am meisten verbreiteten Methoden zur Zahlendarstellung. Um verschiedene Produkte zu vergleichen, können deren Verkaufszahlen jeweils als Serie eingegeben und zusammen in eine Grafik gebracht werden. Eine interessante Abwechslung ist bei den Balkengrafiken durch die sogenannten «Icons« möglich. Icons sind kleine, farbige Grafiksymbole, die mit Hilfe eines speziellen Editors selbst erstellt werden können. Hat man so ein Icon erstellt, lassen sich die Balken aus diesem Symbol bilden, was zur anschaulichen grafischen Darstellung beiträgt. Bild 2 zeigt eine Grafik, die mit Hilfe der Icons erstellt wurde. Leider haben die Icons den Nachteil, daß sie quadratisch sein müssen, was der Kreativität des Benutzers gewisse Fesseln auferlegt.

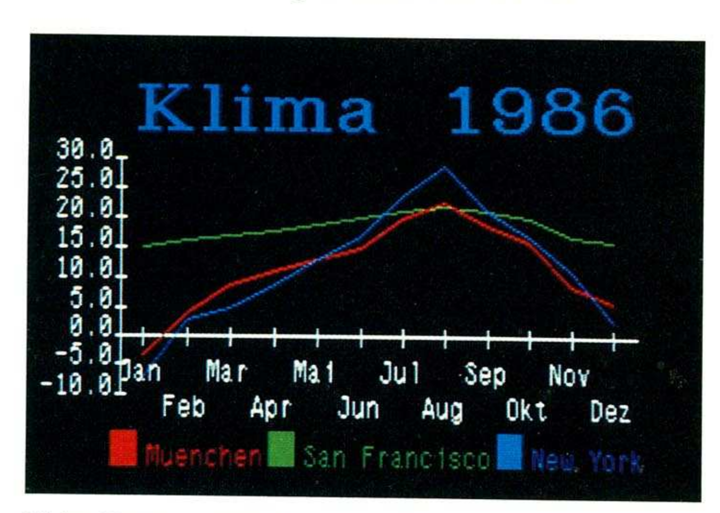


Bild 1. Eine Liniengrafik ohne Gittermuster

Auch bei der Liniengrafik sind einige Variationsmöglichkeiten vorhanden: Die Zahlenwerte lassen sich außer in Form von Linien auch durch ausgefüllte Flächen unterhalb der Linien oder lediglich als Punkte darstellen (Bild 1). Die Liniengrafiken lassen sich auf dem Schirm in die Horizontale bringen, daß heißt um 90 Grad drehen. Einen dreidimensionalen Effekt gibt es hier allerdings nicht. Sehr wirksam ist bei Linien- und Balkengrafiken auch ein Gittermuster, welches »hinter« Linien oder Balken angeordnet werden kann und so die Lesbarkeit der Grafik verbessert. Der Abstand der Gitterlinien ist beliebig wählbar.

Wird schließlich aus den eingegebenen Zahlen eine Kuchengrafik »gebacken«, so rechnet der Computer diese automatisch in Prozentwerte um, mit denen die Kuchenstücke dann beschriftet werden (Bild 3). Eine Kuchengrafik kann entweder flach oder dreidimensional gestaltet werden; zusätzlich kann der Anwender auch noch wählen, ob die Kuchenstücke zusammenliegend, oder durch kleine Zwischenräume getrennt sein sollen.

Nachdem der Computer nun eine bestimmte Grafik erstellt hat, wird man wahrscheinlich noch einiges ändern wollen. Ab und zu rutscht eine Beschriftung in die Grafik hinein oder die Skalenbeschriftung ist (beim dreidimensionalen Effekt) zu weit von den Achsen entfernt. Abhilfe bringt dann die »EDIT«-Funktion, die es ermöglicht, jedes einzelne Element der Grafik zu verschieben, zu löschen, zu kopieren und in der Größe zu verändern, wobei auch mehrere Dinge gleichzeitig verschoben werden können. Die Geschwindigkeit bei der Verschiebung von mehreren Grafikelementen läßt allerdings zu wünschen übrig. Bei Kuchengrafiken ist es möglich, eines oder mehrere Kuchenstücke aus dem Kuchen herauszuschieben, wobei die Beschriftung des Kuchens aber manchmal etwas zerstört wird. Somit kann man das Aussehen der Grafik seinen Wünschen anpassen. Wenn allerdings Text und Grafik sehr nahe beieinander oder gar vollständig aufeinander liegen, gelingt es oft erst nach einigen Versuchen, dem Programm klarzumachen, was man eigentlich bewegen, ko-

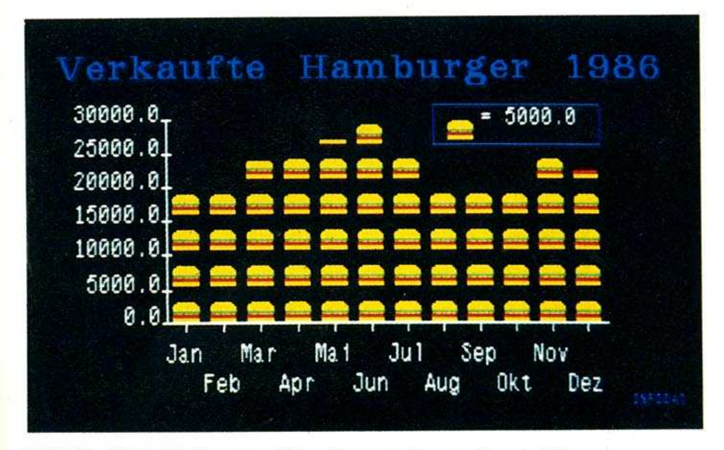


Bild 2. Eine Balkengrafik, die aus Icons besteht

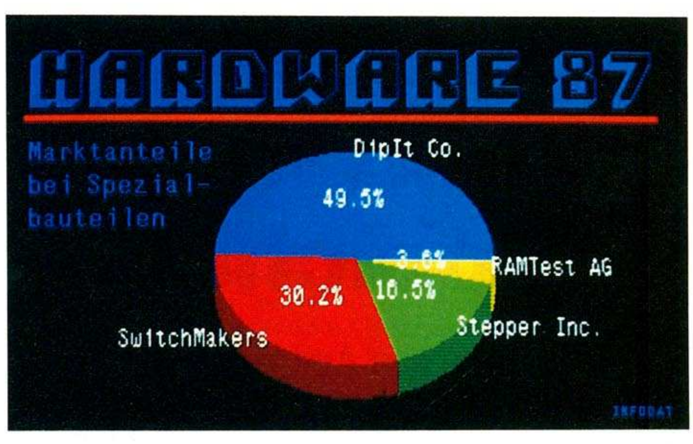


Bild 3. Beschriftungen sind kein Problem

pieren, löschen oder in der Größe verändern will. Ein anderer Nachteil der »EDIT«-Funktion ist, daß bei einer Verschiebung statt dem markierten Objekt nur ein Rechteck mit der Maus zum Zielort bewegt wird, so daß man auch hier mehrere Versuche einkalkulieren sollte. Besser wäre eine Lösung in der Art der Deluxe-Paint-Pinsel gewesen.

Für Kuchengrafiken gibt es eine weitere Einschränkung. Da die Prozentzahlen, die als Beschriftung für die Kuchengrafik dienen, nicht einzeln, sondern nur gleichzeitig verschoben werden können, ist es unmöglich, sie weiter auseinanderzuschieben, was oft wünschenswert wäre. Somit muß der Kuchen mit der Text-Funktion »von Hand« beschriftet werden, was ziemlich umständlich ist.

Die vom Programm verwendeten Farben der grafischen Darstellung können ebenso wie der Bereich der Skalenbeschriftungen frei gewählt werden. Das dafür verwendete Menü hat allerdings einen entscheidenden Haken: wird es aufgerufen, so werden alle bisherigen Veränderungen der Grafik durch »EDIT« rückgängig gemacht, so daß eine einfache Farbänderung (die sich sowieso erst auf eine neu gezeichnete Grafik auswirkt) eine Menge unnötige Arbeit verursachen kann.

Schließlich sollte die Grafik noch ihre »persönliche Note« bekommen. Dazu wurden die Grundfunktionen einiger Zeichenprogramme übernommen, so daß sich in die Grafik noch Linien, Kreise oder selbstgezeichnete Dinge in verschiedenen Pinselstärken einfügen lassen. Auch die oben bereits angesprochenen Icons lassen sich in die Grafik einsetzen. Das bei anderen Aegis-Programmen bereits bewährte Fast-Menü kann auch bei Impact verwendet werden und ermöglicht ein schnelles Anwählen der verschiedenen Zeichenfunktionen. Neu ist aber eine sehr mächtige »Undo«-Funktion, die nicht nur den letzten Befehl zurücknimmt, sondern bei jedem neuen Aufruf von »Undo« wieder einen weiteren.

Text nicht zu vergessen

Zusätzlich gibt es eine sehr vielseitige »Text«-Funktion, mit der in fünf verschiedenen Zeichensätzen in nahezu beliebiger Größe und allen möglichen Sonderfunktionen wie Fettschrift, Schattenschrift, Unterstreichung usw. eine individuelle Beschriftung gestaltet werden kann. Die Freiheit des Benutzers geht hier so weit, daß außer der Buchstabenfarbe selbst sogar die Farbe des Buchstabenschattens einstellbar ist. Leider fehlen der Textfunktion die deutschen Sonderzeichen, was aber bei amerikanischer Software der Normalfall ist. Beim Betätigen der Delete-Taste zum Löschen eines gerade eingegebenen Zeichens ist das Programm etwas zu langsam, da die überschriebene Grafik wieder auf den Bildschirm gebracht werden muß. Abhilfe schafft hier der »Table-Builder«, ein kleiner Texteditor, mit dem Text vor dem Einbau in die Grafik vorbereitet werden kann.

Obwohl das Programm ansonsten relativ schnell ist (etwa bei der Erstellung der Grafiken), so gibt es doch manchmal Geschwindigkeitsprobleme, vor allem wenn irgendwelche Abfrageboxen auf dem Bildschirm erscheinen. Das Programm speichert den durch so eine Box überschriebenen Grafikteil nämlich nicht ab, sondern baut ihn aus den ursprünglichen Grafikdaten wieder auf, was einige Zeit in Anspruch nimmt. Hat die Grafik schließlich ihren letzten Schliff erhalten, kann sie als »Dia« auf Diskette abgespeichert werden. Mit einem zweiten, extra mitgelieferten Programm lassen sich dann viele Grafiken zu einer Dia-Show zusammenstellen. Dabei kann für jedes einzelne Dia die Verweilzeit auf dem Bildschirm sowie die Art des Diawechsels (zum Beispiel einblenden, ausblenden, nach oben oder unten schieben etc.) festgelegt werden. Diese Dia-Show kann dann auf eine Diskette geschrieben werden und läuft vollautomatisch ab. Aegis erlaubt sogar ausdrücklich die Weitergabe des Dia-Show-Programms an andere. Durch den Videoausgang des Amiga läßt sich eine Dia-Show natürlich auch hervorragend zu einem Videofilm verarbeiten.

Das Diskettenhandling ist bei Aegis Impact gut gelöst worden. Sobald dem Programm mitgeteilt wurde, daß das externe Laufwerk eine Datendiskette enthält, liest es von dort die Namen aller auf der Diskette gespeicherten Grafiken ein. Somit kann ein schneller Zugriff auf die Diskette ohne umständliches Lesen des Directory erfolgen. Der Name der gewünschten Grafik muß dann mit Hilfe des Menüpunktes »Directory« ermittelt und beim Laden über die Tastatur eingegeben werden.

Obwohl das Programm laut der Packung zwei Diskettenlaufwerke benötigt, ist auch das Arbeiten mit einem Laufwerk möglich. Da die Anleitung dazu einige etwas irreführende Tips gibt, soll die Benutzung des Programms mit einem Laufwerk kurz erklärt werden. Dazu wird im CLI folgendes eingegeben:

```
copy impact!:fonts to ram:
makedir data:fonts
copy ram: to data:fonts all
```

Dabei wird angenommen, daß die Datendiskette »data« heißen soll. Nach dem Laden des Programms die Systemdiskette herausnehmen, die Datendisk einlegen und den Menüpunkt »Drawer« im Project-Menü anklicken. Dort sofort die OK-Box anklicken. Dadurch wird das Inhaltsverzeichnis der Datendisk in den Speicher gebracht. Vor dem Verlassen von Impact muß dann wieder die System-Diskette eingelegt werden. Bei der Slideshow wird nach demselben Muster verfahren.

Das umfangreiche englische Anleitungsbuch enthält eine Einführung in Impact für Anfänger und genaue Beschreibungen der einzelnen Funktionen. Es ist mit S/W-Bildschirmfotos illustriert und grafisch gut gelungen. Ein Problem ist allerdings, daß das Anleitungsbuch auf ein Dia-Show-Programm bezogen ist, das Teil des Impact-Programmes ist, was bei der getesteten Impact-Version nicht der Fall war. Zwar weist ein Zettel in der Packung auf diesen Fehler hin, trotzdem werden hier Probleme geschaffen, wo keine sind.

Insgesamt ist Aegis Impact ein gutes Programm, dessen grafisch gelungenen Ergebnisse auch über manche kleinen Macken hinwegtrösten. Die Geschwindigkeit ist nicht immer auf Amiga-Niveau, teilweise ist die Langsamkeit des Programmes ärgerlich. Bei Impact wurde übrigens auf einen Kopierschutz verzichtet.

(Andreas Lietz/jk)

Anbieter (Adressen siehe Marktübersicht): Alphatron, Atalantis, B. Soltau Elelctronics KG, DTM, Instant Replay, Interplan, ITC, Philgerma GmbH, Softwareland, Soyka Datentechnik, SWS Computersysteme, UBM-Drecker, Video Loft Film, Medien-Center
Preis: 199 bis 629 Mark

Schneller durch mehr Speicher

Die RAM-Karte des Amiga 2000 kann bis zu 1 MByte ausgebaut werden, doch leider hat sie noch einen kleinen Fehler. Mit der Beseitigung dieses Fehlers und dem Einbau des MC68010 wird der Amiga dann wirklich schneller. Außerdem helfen wir Ihnen beim Einbau eines zweiten Floppy-Laufwerks.

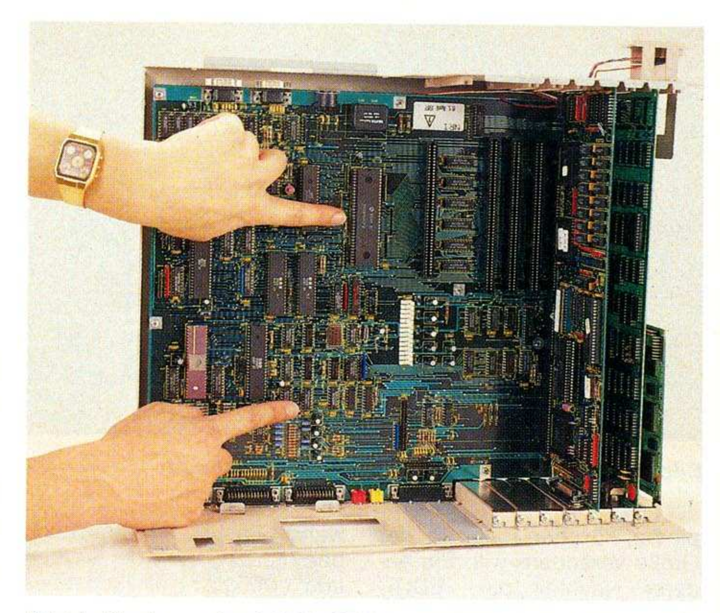


Bild 1. Die Lage der 68000-CPU (oben) und des Jumpers für das zweite interne Laufwerk (unten)

Drei kleine Änderungen am Amiga 2000 machen ihn mit mehr Speicher schneller. Bereits im 68000er Sonderheft 12/86 wurde die Umrüstmöglichkeit des Amiga 1000 auf den 68010-Prozessor beschrieben. Dieser Prozessor ist inzwischen in akzeptable Preisklassen »gerutscht«, schon für etwa 75 Mark erhält man ihn im Handel. Wir haben die Umrüstung des Amiga 2000 auf den 68010 durchgeführt und einige Tests unternommen, die den realen Geschwindigkeitsvorteil für den Anwender zeigen sollen.

Zunächst ein paar Worte zum Prozessor: Der 68010 ist vollständig pinkompatibel zum 68000. Zur Umrüstung des Amiga 2000 ist, wie für den Amiga 1000, der MC68010L8, der für eine Taktfrequenz bis 8 Megahertz geeignet ist, notwendig. Einige Schleifen und Befehle werden von ihm bis zu 80 Prozent schneller abgearbeitet als vom 68000. Dies hört sich beeindruckend an, interessanter ist für den Anwender jedoch der tatsächliche Zeitgewinn bei der Ausführung »normaler« Programme. Sie bestehen, wie man sich leicht vorstellen kann, nicht nur aus den Befehlen, bei denen der neue Prozessor Vorteile bietet.

Wer seinen Amiga auf den 68010 umgerüstet hat, muß zunächst seine Software auf dem »getunten« Computer zum Laufen bringen. Nicht alles läuft nämlich ohne Vorkehrungen auf dem 68010: Der Prozessorbefehl »MOVE SR, <ea>« darf beim 68010 (und »höheren« Typen der 68000er-Serie) nur unter bestimmten Bedingungen aufgerufen werden. Nimmt man hierauf keine Rücksicht, stürzt der Computer sehr wahrscheinlich ab. Zum Glück gibt es jedoch auf der Public Domain-Fish-Disk Nummer 18 ein kleines Programm namens »DeciGEL«. Ist dieses Programm geladen, wird der unerwünschte Befehl umgeleitet und die Software läuft wieder. Es empfiehlt sich also, DeciGEL auf allen Bootdisketten spätestens dann, wenn der »Guru« einmal »meditiert« hat, ins C-Directory zu legen und in der s/startup-sequence aufzurufen.

Schneller mit dem MC68010

Nun endlich zum wichtigsten Teil des Artikels für Geschwindigkeitsbegeisterte und potentielle Umrüster: dem Laufzeitvergleich. Zur Verfügung standen uns zwei gleiche Amiga 2000, wovon einer mit dem 68010 ausgerüstet war. Als »Benchmarks« wählten wir verschiedene Programme:

Eine Primzahlenbestimmung bis 64000, eine Integer- sowie eine Floating-Point Additions-/Subtraktions-/Multiplikations-Divisionsschleife mit 100000 Durchläufen (intbench.c und floatbench.c, Listing 1), geschrieben in C.

In Tabelle 1 haben wir die Ergebnisse zusammengefaßt. Das DeciGEL-Programm verursachte keine meßbaren Laufzeitunterschiede, es schadet also nichts, wenn man es vorsichtshalber von allen Bootdisketten »zuladen« läßt.

Die Umrüstung des Amiga 2000 ist, im Vergleich zu der des 1000ers, einfach, da das Innenleben des Amiga 2000 leichter zugänglich ist. Hier muß jedoch bemerkt werden, daß keine Verantwortung für eventuell bei der Umrüstung oder Änderung nach den folgenden Beschreibungen entstehende Fehler oder Beschädigungen übernommen werden kann. Nach dem Ziehen des Netzsteckers entfernt man die mit fünf Schrauben befestigte Gehäuseabdeckung (je zwei Schrauben an der rechten und linken Seitenwand unten, eine Schraube oben in der Mitte der Rückwand). Bei allen folgenden Arbeiten im Computer sollte man daran denken, daß eine eventuell noch gültige Garantie erlöschen kann. Obwohl die TTL-Bausteine heutzutage nicht mehr so empfindlich sind, ist darauf zu achten, daß keine statische Ladung über die Elektronik des Computers

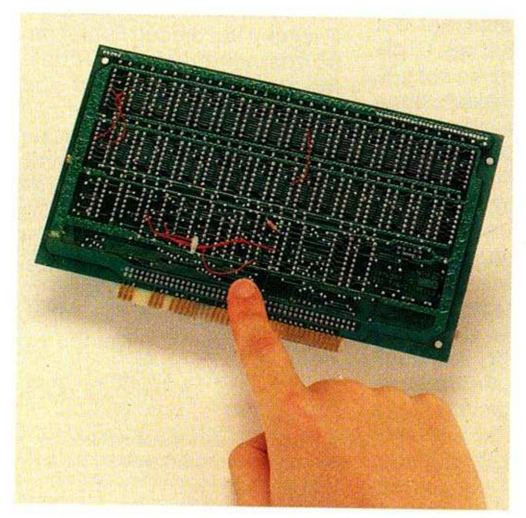


Bild 2. Dieses Kabel muß umgelötet werden, um die Wait-States zu beseitigen

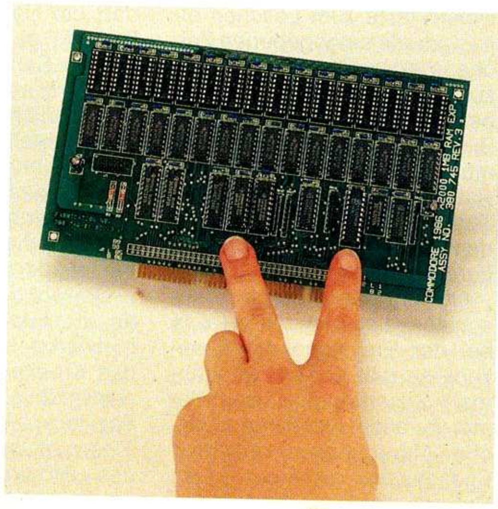


Bild 3. Zwischen diesen Bausteinen liegt die Drahtbrücke

abfließt. Wer sich hier nicht ganz sicher ist, oder noch keine Erfahrung mit dem Umgang mit »vielbeinigen« ICs hat (der 68000/68010 hat deren immerhin 64!), sollte die Umrüstung besser von seinem Commodore-Fachhändler vornehmen lassen.

Der Prozessor liegt beim Amiga 2000 halb verdeckt unter der Brücke über der Hauptplatine, die die Laufwerke und das Netzteil trägt. Wenn kein 5¼-Zoll-Laufwerk eingebaut ist, kann man sich das einfache Ausbauen dieser Brücke sparen. Nach dem Herausziehen der werksmäßig mit 512 KByte bestückten Speichererweiterungs-Platine, die im sogenannten »Prozessor-Slot« gleich links neben den Laufwerken steckt, liegt der Prozessor fast schon greifbar vor uns. Um ihn jetzt beiderseitig mit einem flachen, nicht zu schmalen Schraubenzieher vorsichtig heraushebeln zu können, muß man nur noch das vom Netzgerät kommende Stromversorgungskabel an der Hauptplatine abziehen (Verriegelungsleiste leicht zurückbiegen). Von hinten ist der Prozessor dann frei zu erreichen, von vorne kommt man an ihn durch den Ausschnitt im leeren 5¼-Zoll-Laufwerksgehäuse heran. Ist er losgehebelt, nimmt man ihn vorsichtig, ohne die Beinchen zu berühren, heraus und legt den »Neuen« locker auf den Sockel. Achtung: Die Markierungskerbe muß zur Gerätevorderseite zeigen! Bevor man den 68010 in seinen zukünftigen »Stammplatz« (Bild 1 oben) eindrückt, sollte man sich vergewissern, daß er nicht etwa um ein oder mehrere Beinpaare versetzt auf dem Sockel sitzt und jedes Bein in seinem Loch liegt. Ein danebenliegendes Bein wird beim Eindrücken schnell zum abgebrochenen, was für einen Mikroprozessor üblicherweise »tödlich« ist. Ist der 68010 sauber eingedrückt, steckt man das Platinen-Stromversorgungskabel vom Netzteil wieder auf seinen Stecker. Auch hier ist ein Kontrollblick angebracht, ob es nicht etwa versetzt aufgesteckt wurde. Auch ein komplettes Vergessen dieses Kabels kann beim Einschalten zum Verlust des teuren Schaltnetzteiles führen! Nach dem Einstecken der RAM-Erweiterung (bestückte Seite zum Laufwerk!) und dem Wiederaufsetzen der Abdeckung ist der Amiga wieder betriebsbereit. Sollte er sich jetzt nicht wie gewohnt beim Einschalten »melden«, so ist das Netzkabel nicht eingesteckt, der neue Prozessor defekt — dies kann man durch »Zurücktauschen« mit dem Originalexemplar feststellen — oder man hat etwas beschädigt. In diesem — hoffentlich nicht vorkommenden — Fall hilft nur der Gang zum Fachhändler.

Schneller ohne Waitstates

Bei unseren Tests stießen wir auf eine zunächst erstaunliche, für die A-2000-Entwickler nicht gerade ruhmreiche Tatsache: Manche unserer Testprogramme liefen plötzlich bis zu 20 Prozent schneller in derselben Computer-Konfiguration! Nach einigem Rätseln stellten wir fest, daß der schnellere Programmablauf immer dann stattfand, wenn das sogenannte »Fast«-Memory, also der auf der Steckkarte untergebrachte Speicher, belegt war und unsere Programme auf das »Chip«-Memory ausweichen mußten. Im ersten Moment erscheint es paradox, daß der Computer schneller werden soll, wenn der Speicher voll ist. Die Erklärung für dieses Verhalten ist auf der serienmäßig eingesteckten Speichererweiterungskarte zu suchen: Sie erzeugt ganz offensichtlich »Wait-States«. Das sind Leer-Taktzyklen, die der Prozessor warten muß, bis die Werte am Datenbus anliegen. Dies ist unserer Ansicht nach völlig unnötig und zeugt von einer hektischen Entwicklungsarbeit bei der Auslegung der Speichererweiterung, da die Speicherbausteine des »Fast«-Memory auf der Einsteckkarte die gleichen wie die des »Chip«-Memory auf der Hauptplatine sind. Daß es möglich ist, Speichererweiterungen ohne Wait-States für den Amiga zu bauen, zeigen verschiedene 2-MByte-Ansteckerweiterungen für den Amiga 1000.

Nun zu unserer Lösungsmöglichkeit des RAM-Problems, das zumindest bei den früh (im März/April dieses Jahres) ausgelieferten Amiga 2000 auftritt. Hier ist etwas Löt-

arbeit erforderlich. Diese sollte verständlicherweise nur von jemandem ausgeführt werden, der schon etwas Erfahrung damit hat. Auf der Rückseite der 512-KByte-RAM-Erweiterung (Bild 2) gibt es noch sechs Drahtbrücken. Von der Drahtbrücke, die Pin 9 von U 16 mit Pin 12 von U 13 (Bild 3) verbindet, wird der letztgenannte Anschluß auf Pin 17 von U 13 umgelegt. Jetzt ist die RAM-Erweiterungskarte offensichtlich im vom Entwickler vorgesehenen Zustand. In unserer Vergleichstabelle haben wir die neuen, recht ansehnlichen Laufzeiten unserer Testprogramme aufgeführt. Die prozentuale Beschleunigung des Computers durch die Änderung auf der RAM-Karte ist übrigens praktisch unabhängig vom eingesetzten Prozessor. Sie lohnt sich also auch für diejenigen, die mit dem 68000 weiterarbeiten wollen. Ein weiterer Erfolg dieser Maßnahme ist, daß »zeitkritische« Programme, wie zum Beispiel das Deluxe-Music-Construction-Set, nun nicht mehr auf dem 2000er »stolpern«.

Nachdem wir bei mehreren Amiga 2000 die Änderung vorgenommen haben, mußten wir jedoch feststellen, daß sich einer der Computer standhaft weigerte, problemlos nach dem Umbau zu arbeiten. Wir konnten noch nicht feststellen, warum gerade dieser Amiga 2000 »streikte«. Entweder beruhte der Fehler auf Fertigungstoleranzen bei den RAM-Bausteinen oder der eingebauten PC-Karte. Abhilfe schafft aber ein Zurücklöten des Kabels, um wieder die alte Geschwindigkeit zu erreichen.

Ganz einfach als ärgerlich muß man die Wait-States auf der RAM-Karte bezeichnen. Sie kosteten 14 bis 22 Prozent der Rechenzeit bei unseren Tests.

Damit zehren sie den möglichen Geschwindigkeitsgewinn des Fast-RAM völlig auf. Hier bleibt nur zu hoffen, daß Commodore schnellstmöglich Änderungen vornimmt, um dem Fast-RAM zu einer Rechtfertigung seines Namens zu verhelfen, da sich sicherlich nicht jeder Amiga-Besitzer als Hardware-Bastler betätigen möchte oder kann. Wem nun der Amiga immer noch nicht schnell genug ist, dem bleibt noch die Möglichkeit, durch geschicktes Setzen der Prioritäten mit dem Settaskpri-Befehl des CLI unerwünschte Nebenaktionen des Prozessors während des Programmablaufs zu unterbinden. Wenn die geschilderten Möglichkeiten der Geschwindigkeitssteigerung ausgenutzt sind, wird die Arbeit mit dem Amiga 2000 noch genußvoller, als sie ohnehin schon ist.

Unproblematisch, auch nach Durchführung der beschriebenen Änderungen, ist übrigens die Speichererweiterung des Amiga 2000 auf 1,5 MByte. Auf der RAM-Erweiterungskarte im Prozessor-Slot, die bei der Auslieferung mit 512 KByte RAM bestückt ist, sind bereits Stecksockel für weitere 512 KByte eingelötet. Diese müssen mit 16 257-KBitx1-(Nibble Mode) dynamischen RAM-Chips mit einer Zugriffszeit von maximal 150 ns bestückt werden. Geeignet sind zum Beispiel NEC D41257C-15 oder kompatible Typen anderer Hersteller. Damit der Amiga den neuen Speicher erkennt und anspricht, müssen die Jumper unterhalb der RAM-Bausteine auf der Karte umgesteckt werden. Im Lieferzustand sind die drei mittleren Pinpaare des Jumperfeldes verbunden. Für die neue Speicherkonfiguration sieht der »Steckplan« (von links nach rechts) wie folgt aus: 0 1 0 0 1 (0 = offen, 1 = verbunden). Der Preis für diese, besonders für die Benutzer umfangreicher CAD- und Malprogramme lohnenswerte Erweiterung liegt bei 160 bis 190 Mark.

Abschließend geben wir noch Hilfestellung beim Einbau des zweiten Laufwerks (df1) für den Amiga 2000. Leider liefert Commodore keine Einbauanleitung bei diesem einzeln verpackten Floppylaufwerk mit. Man erhält dazu nur einen kleinen Plastikbeutel, in dem sich Befestigungsschrauben und ein Brückenstecker befinden, von dem man nicht so ohne weiteres weiß, wohin dieser gehört.

Um das zweite interne Laufwerk einzubauen, muß zuerst die Befestigungsplatte, auf der sich das erste Laufwerk befindet, abgebaut werden. Merken Sie sich dabei unbedingt die Lage aller Stecker und Steckkontakte! Anschließend entfernt man das komplette Netzteil, welches an der Gehäusevorder- und -rückseite mit zwei beziehungsweise vier Schrauben befestigt ist. Nach Abziehen der Steckverbinder von der Hauptplatine kann das komplette Schaltnetzteil aus dem Amiga entnommen werden. Unter dem Netzteil findet sich ein Jumper (Bild 1 unten), der mit dem mitgelieferten Brückenstecker überbrückt wird. Erst dadurch kann das zweite Laufwerk überhaupt erkannt und angesprochen werden.

Bauen Sie das Netzteil wieder ein und befestigen alle Steckverbinder vom Schaltnetzteil zur Hauptplatine.

Schrauben Sie anschließend das erste Laufwerk vom Halteblech ab und legen es auf die Oberseite. Auf dem zweiten Laufwerk sind leider zwei Jumper falsch gesteckt. Verbinden Sie deshalb alle Jumper auf dem zweiten Laufwerk so, wie sie im ersten gesteckt sind, bis auf den Jumper zur Laufwerksadresse. Sie sehen auf der Jumperreihe vom zweiten Laufwerk die Bezeichnungen »df0«, »df1«, »df2« und etwas weiter hinten »df3«. Damit das neue Laufwerk als »df1« angesprochen werden kann, muß der Jumper von »df0« auf »df1« gesteckt werden.

Sie können nun wieder beide Diskettenlaufwerke mit dem Halteblech verschrauben und das komplette Gestell einbauen. Als letzte Handlung sind noch die Strom- und Busstecker auf die Laufwerke aufzustecken, wobei darauf geachtet werden muß, daß die einzelnen Leitungen in die gleiche Richtung zeigen. Nach Aufsetzen des Gehäuseblechs und Verschrauben desselben steht Ihnen ein Amiga 2000 zur Verfügung, der nun über zwei eingebaute 3½-Zoll-Diskettenlaufwerke verfügt.

(J. Ewald/K. Müller-Holthusen/rb/dm)

Mehr Speicher auf der RAM-Karte

Was bleibt nun als Fazit der »Beschleunigungsbetrachtungen«? Mit Sicherheit ist der Kauf und Einbau des 68010-Prozessors für alle, die rechenintensive Programme auf ihrem Amiga laufen lassen, interessant. Ein »Wunderprozessor« ist er, im Vergleich zum 68000 jedoch nicht, wohl aber eine attraktive Alternativlösung. Immerhin kann man den Computer mit knapp 2 Prozent Mehraufwand je nach Einsatzzweck um 2 bis 9 Prozent beschleunigen. Bei speziellen Anwendungen läßt sich sicher auch noch mehr herausholen.

Geschwindigkeitstabelle: (Zeiten in Minuten:Sekunden)

Amiga 2000	»Fast«-RAM	Standard	modifizierte Karte	
Prozessor	68000	68010	68000	68010
Programm:				
primsped.c	0:04,55	0:04,45	0:03,62	0:03,55
intbench.c	0:26,00	0:24,50	0:21,80	0:20,45
floatbench.c	0:54,00	0:51,00	0:47.70	0:44,70

```
#include <stdio.h>
long a,b;
/* Für Fließkommazahlentest muß hier
»float a,b;« stehen */
main()
{
          printf("Start\n");
          for(a=1;a<100000;a++)
          {
                    b=a+a;
                    b=b*a;
                    b=b/a;
                    b=b-a;
          }
          printf("\nFertig!\n");
}
```

Listing 1. Einfacher Benchmarktest mit Integer- beziehungsweise Float-Zahlen und den vier Grundrechenarten

Theorie und Praxis des Druckens mit dem Amiga

Wußten Sie, daß der Amiga ein besonders raffiniertes System verwendet, um einen Drucker anzusprechen? Lesen Sie, wie man beim Amiga Anpassungsprobleme vermeidet, wie man Drucker programmiert und welche Farbdrucker für die Grafikmaschine Amiga geeignet sind.

Wysiwyg (What you see is what you get) ist eine der am meisten verwendeten Abkürzungen in der Computertechnik. Doch drehen wir einmal die Frage um und prüfen, ob der Ausdruck am Drucker die gleiche Qualität hat wie die Darstellung am Bildschirm.

Das Betriebssystem des Amiga kennt drei »Geräte« (Devices) zum Ansprechen eines Druckers. Diese Ausgabegeräte werden mit »PAR:«, »SER:« und »PRT:« bezeichnet. Mit »PAR:« wird die parallele Schnittstelle ausgewählt, die alle Drucker mit der Centronics-Schnittstelle ansteuern kann. »SER:« wählt die serielle Schnittstelle des Amiga aus. Diese beiden Ausgabegeräte bezeichnen also direkt vorhandene Hardware-Einrichtungen des Amiga. Die eigentliche Druckerschnittstelle ist aber »PRT:«. Sie wird auch als Printer-Device bezeichnet und ist als Teil des Betriebssystems keiner Hardware-Einrichtung zugeordnet. Das Printer-Device stellt vielmehr ein Programm (innerhalb des Betriebssystems) dar, das für die Ansteuerung des jeweils aktivierten Druckertreibers eines Programms zuständig ist. Für das Printer-Device gelten deshalb die Einstellungen, die mit Preferences getroffen werden. Es wird automatisch die eingestellte Schnittstelle (parallel oder seriell) gewählt. Mit der Wahl des Druckertreibers weiß das Printer-Device, um welchen Druckertyp es sich handelt und welche Fähigkeiten der Drucker besitzt.

Für den Textausdruck enthält jeder Druckertreiber Informationen über die Anzahl der maximal möglichen Zeichen pro Zeile und die Anzahl der möglichen Zeichensätze. Für den Textausdruck gelten spezielle Escape-Sequenzen (Tabelle 1). Diese werden durch das Printer-Device in die entsprechenden Steuercodes für den Drucker umgewandelt. Die Zeichenfolge ESC[4m schaltet Unterstreichen ein, ESC[2w schaltet die Schriftart Elite ein. Insgesamt kennt das Printer-Device über 70 verschiedene Escape-Sequenzen, die in der Tabelle 1 zusammengefaßt sind. Die erste Spalte gibt den Namen des Kommandos wieder, der von C-Programmierern verwendet werden kann. Die zweite Spalte enthält die Escape-Sequen-

Kommando Name	Escape Sequence	Funktion
aRIS	ESCc	Rücksetzen
aRIN	ESC#1	Initialisieren
aIND	ESCD	Zeilenvorschub
aNEL	ESCE	Wagenrücklauf, Zeilenvorschub
aRI	ESCM	Zeilenvorschub rückwärts
aSGR0	ESC[0m	Normaler Zeichensatz
aSGR3	ESC[3m	Schrägschrift (italics) ein
aSRG23	ESC[23m	Schrägschrift (italics) aus
aSGR4	ESC[4m	Unterstreichen ein
aSGR24	ESC[24m	Unterstreichen aus
aSGR1	ESC[1m	Bold-Schriftart ein
aSGR22	ESC[22m	Bold-Schriftart aus
aSFC	ESC[xxm	(xx= 30-39) Wählt Vordergrundfarbe
aSBC	ESC[xxm	(xx= 40-49) Wählt Hintergrundfarbe
aSHORP0	ESC[0w	Normale Zeichenbreite
aSHORP2	ESC[2w	Schaltet Elite-Schriftart ein
aSHORP1	ESC[1w	Schaltet Elite-Schriftart aus
aSHORP4	ESC[4w	Schaltet Schmalschrift ein
aSHORP3	ESC[3w	Schaltet Schmalschrift aus
aSHORP6	ESC[6w	Schaltet Breitschrift ein
aSHORP5	ESC[5w	Schaltet Breitschrift aus
aDEN6	ESC[6"z	Schaltet Schattenschrift ein
aDEN5	ESC[5"z	Schaltet Schattenschrift aus
aDEN4	ESC[4"z	Schaltet Doppeldruck ein
aDEN3	ESC[3"z	Schaltet Doppeldruck aus
aDEN2	ESC[2"z	Schaltet NLQ ein
aDEN1	ESC[1"z	Schaltet NLQ aus
aSUS2	ESC[2v	Schaltet Tiefstellen ein
aSUS1	ESC[1v	Schaltet Tiefstellen aus
aSUS4	ESC[4v	Schaltet Hochstellen ein
aSUS3	ESC[3v	Schaltet Hochstellen aus
aSUS0	ESC[0v	Schaltet auf normale Zeile zurück
aPLU	ESCL	Stelle auf untere Zeilenhälfte
aPLD	ESCK	Stelle auf obere Zeilenhälfte
aFNTO	ESC(B	US Zeichensatz
aFNT1	ESC(R	Französischer Zeichensatz
aFNT2	ESC(K	Deutscher Zeichensatz
aFNT3	ESC(A	US Zeichensatz
aFNT4	ESC(E	Dänischer Zeichensatz
aFNT5	ESC(H	Schwedischer Zeichensatz
aFNT6	ESC(Y	Italienischer Zeichensatz
aFN7	ESC(Z	Spanischer Zeichensatz
aFNT8	ESC(J	Japanischer Zeichensatz
aFNT9	ESC(6	Norwegischer Zeichensatz
aFNT10	ESC(C	Dänischer Zeichensatz II
aPROP2	ESC[2p	Proportionalschrift ein
aPROP1	ESC[1p	Proportionalschrift aus
aPROP0	ESC[0p	Proportionalschrift löschen
aTSS	ESC[n E	Setzt den Abstand für Proportionalschrift
aJFY5	ESC[5 F	Automatische Justierung linksbündig
aJFY7	ESC[7 F	Automatische Justierung rechtsbündig
aJFY6	ESC[6 F	Automatische Justierung mittig
aJFY0	ESC[0 F	Ausschalten der automatischen Justierung
aJFY3	ESC[3 F	Zeichenabstand für Justierung
aJFY1	ESC[1 F	Wort füllen (automatische Zentrierung)
aVERP0	ESC[0z	1/8 Zoll-Zeilenabstand
aVERP1	ESC[1z	1/6 Zoll-Zeilenabstand
aSLPP	ESC[nt	Seitenlänge n Zeilen
aPERF	ESC[nq	Perforationssprung wirksam (n 0)
aPERF0	ESC[0q	Perforationssprung unwirksam
aLMS	ESC#9	Linken Rand setzen
aRMS	ESC#0	Rechten Rand setzen
aTMS	ESC#8	Oberen Rand setzen
aBMS	ESC#2	Unteren Rand setzen
aSTBM	ESC[xx;yyr	Oberen und unteren Rand setzen
aSLRM	ESC[xx;yys	Linken und rechten Rand setzen
aCAM	ESC#3	Ränder löschen
aHTS	ESCH	Horizontalen Tabulator setzen
aVTS	ESCJ	Vertikalen Tabulator setzen
aTBC0	ESC[0g	Horizontalen Tabulator löschen
aTBC3	ESC[3g	Löschen aller horizontalen Tabulatoren
aTBC1	ESC[1g	Vertikalen Tabulator löschen
aTBC4	ESC[4g	Löschen aller vertikalen Tabulatoren
aTBCALL	ESC#4	Löschen aller Tabulatoren
aTBSALL	ESC#5	Setzen voreingestellter Tabulatoren
aEXTEND	ESC[n"x	Erweiterungs-Kommandos

Tabelle 1. Der Drucker-Device-Treiber des Amigas versteht nur diese Befehle, ist dadurch aber besonders flexibel

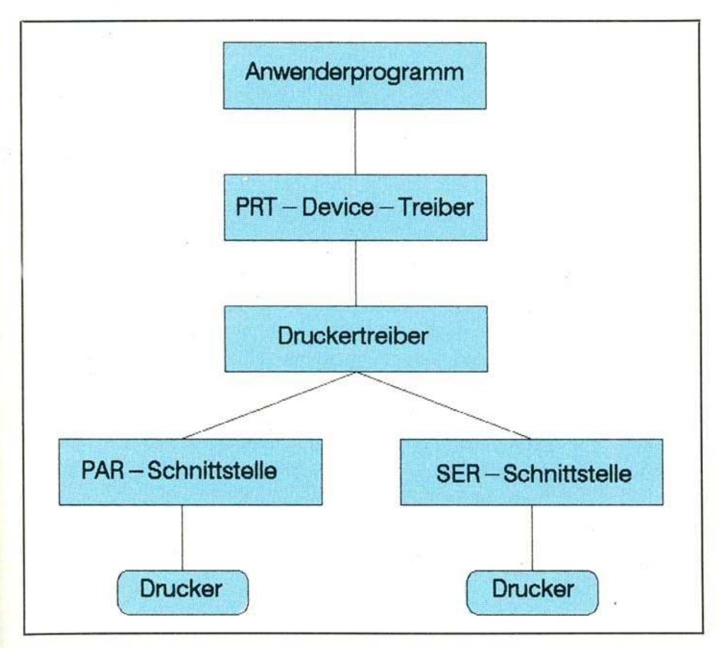


Bild 1. Schematischer Ablauf der Druckerausgabe über das »Printer.Device« beim Amiga

```
OPEN "PRT:" FOR OUTPUT AS #1
ulon$= CHR$(27)+"[4m"
uloff$=CHR$(27)+"[24m
elon$= CHR$(27)+"2w"
eloff$=CHR$(27)+"[1w"
conon$=CHR$(27)+"[4w"
conoff$=CHR$(27)+"[3w"
SLRM$ =CHR$(27)+"[40;50s"
CAM$= CHR$(27)+"#3"
PRINT#1,SLRM$
PRINT#1,"normaler Text";
PRINT#1,ulon$;" unterstrichen"uloff";
PRINT#1," wieder normale Schrift"
PRINT#1,elon$;"ELITE-Schrift";eloff$;"
wieder normale Schrift
PRINT#1,CAM$
PRINT#1,conon$;"Komprimier-
ter Druck";conoff$;" und wieder aus"
CLOSE#1
```

Listing 1. Zusammenfassung von Beispielen einiger Steuercodes, die von den meisten Druckern unterstützt wurden

zen, die auch in Basic-Programmen verwendet werden kann. Das Listing 1 zeigt die Verwendung einiger Steuercodes, die von den meisten Druckern unterstützt werden. Die Übergabe eines Befehls an den Drucker über das Printer-Device funktioniert beim Amiga also wie in Bild 1 dargestellt. Das Steuerprogramm (zum Beispiel ein Basicprogramm) übergibt den Befehlsstring (zum Beispiel ESC[4m) an das Printer-Device. Das Printer-Device übergibt den Befehlsstring wiederum an den in Preferences eingestellten Druckertreiber weiter. Der Druckertreiber übersetzt den erhaltenen Befehlsstring in einen Befehlsstring gleicher Bedeutung in der Befehlssprache des Druckers um (aus ESC[4m wird der Epson-Befehl ESC "-" 1 für Unterstreichen ein). Dieser für den Drucker nun »verständliche« Befehl wird dann über die eingestellte Schnittstelle (»SER:« oder »PAR:«) an den Drucker ausgegeben.

Grafikdruck mit »PRT:«

Das Printer-Device enthält eine Funktion zum Ausdruck von Bildern. Diese Funktion wird sowohl von den Malprogrammen wie Deluxe Paint als auch von den Desktop Publishing-Programmen wie Pagesetter verwendet. Damit diese Funktion ausgeführt werden kann, muß der Druckertreiber einen speziellen Programmteil zum Ausdruck von Grafik bereitstellen. Durch die Daten des Druckertreibers ist dem Printer-Device bekannt, ob es sich bei dem ausgewählten Drucker um einen Schwarzweiß- oder einen Farbdrucker handelt. Es wird auch mitgeteilt, wieviel Spalten pro Zeile gedruckt werden können, wieviel Nadeln der Drucker besitzt und welchen Abstand die Punkte in X- und Y-Richtung haben. Aus diesen Daten berechnet das Printer-Device, wie breit das Bild werden soll und sendet dem Druckertreiber die Anzahl der verwendeten Spalten. Der Druckertreiber berechnet dann die Anzahl der Zeichen, die pro Druckzeile erforderlich ist. Es wird vom Druckertreiber immer eine komplette Zeile an den Drucker gesendet. Die Größe des Puffers für diese Zeile setzt sich aus den Steuercodes und den eigentlichen Druckzeichen zusammen. Für einen Epson-Drucker wäre das, bei einer Auflösung von 480 Spalten pro Zeile, eine Puffergröße von 486 Zeichen. Für einen Farbdrucker wie den CP6 von NEC (MPS 2000C) hätte die Puffergröße bei der höchsten Auflösung von 180 x 180 Punkten pro Zoll die stattliche Größe von 17320 Zeichen. Danach wird der Puffer initialisiert und das Printer-Device berechnet die Punkte, die beim Drucker gesetzt werden müssen. Die Koordinaten der Punkte werden einzeln an den Druckertreiber weitergegeben, der dann das entsprechende Bit im Puffer setzt. Da dieser Programmteil für jeden Punkt und jede Farbe des Puffers aufgerufen werden muß, ist die Zeit für eine Zeile von der gewählten Grafikdichte abhängig. Für einen Epson-Drucker mit einer Auflösung von 60 Punkten pro Zoll (480 Spalten pro Zeile) wird die Routine maximal 3840mal pro Zeile aufgerufen. Bei einem NEC CP6 (MPS 2000C) kann dieser Programmteil maximal 138240mal aufgerufen werden und das braucht seine Zeit. Sind alle erforderlichen Punkte im Puffer gesetzt, so wird der Puffer an den Drucker übertragen und die Zeile wird gedruckt. Am Ende eines Bildes wird der Pufferspeicher wieder freigegeben. Durch dieses Verfahren ergibt sich für einen Epson-Drucker eine relativ akzeptable Druckzeit von zirka 5 Minuten, für einen 24-Nadel-Drucker verlängert sich diese Zeit auf bis zu eine halbe Stunde für ein Bild. Daß es auch anders geht, zeigt Bild 2, das mit einem NEC CP 6 (MPS 2000C) und einem speziellen Druckprogramm (mit dem auch die Farben für den Ausdruck verändert werden können) erstellt wurde. Der Ausdruck für dieses Bild hat nicht länger als sieben Minuten gedauert.

Der Vorteil des Printer-Device liegt darin, daß beim Aufruf der Dumproutine (Hardcopyroutine des Betriebssystems) angegeben werden kann, wie groß das Bild auf dem Drucker werden kann. Dabei kann die Größe entweder in Punkten oder in absoluten Zoll- oder Millimeter-Werten angegeben werden.

Das relativ schlechte Ergebnis beim Ausdruck des Programms Pagesetter mit einem 8-Nadel-Drucker ist übrigens darauf zurückzuführen, daß die Bildschirmdump-Routine verwendet wird und keine direkte Ansteuerung des Druckers erfolgt. Dadurch könnte eine Druckqualität erzielt werden, die der anderer Programme auf anderen Systemen in nichts nachsteht. Vielleicht gelingt es auch Commodore, das Drucker-Device schneller und besser zu machen, um einen qualitativ hochwertigen Ausdruck in angemessener Zeit zu erhalten.

Einstellungen mit Preferences

Die mit dem Programm Preferences gewählten Einstellungen haben sowohl bei Textausdruck als auch bei einem Bildschirmdump Wirkung. Bei dem Druckertreiber für den



Bild 2. Solche Grafiken lassen sich nur mit einem speziellen Druckprogramm und einem NEC P 6 erreichen.


Markt&Technik
68000er
DAS MAGAZIN DER NEUEN COMPUTER-GENERATION
FEBRUAR 1987
DM 7,-
2/87
Atari ST
Amiga
Grafik-Sensation Deluxe Paint II
QL
Große Assembler-Parade
Mac
kreativ!
Interview mit »Axel F.« – Komponist Harold Faltermeyer
Viele anspruchsvolle Listings für Atari ST, Amiga und QL
AUFBRUCH IN EINE NEUE DIMENSION
mit »68000er«, dem Magazin der neuen Computer-Generation
▶ Programmiersprachekurse für Basic, C, Modula und Assembler.
▶ Bauanleitungen für professionelle Hardware-Erweiterungen.
▶ Spiele-Spaß und -Spannung auf höchstem Niveau.
Ihre hot-line zur Spitzentechnologie von AtariST, Amiga, Macintosh und Sinclair QL.
Das »68000er«-Magazin erscheint
jeden Monat neu!
POSTER & GUTSCHEIN
KOSTENLOS FÜR SIE
84 mal 60 Zentimeter High-Tech-Szene erwarten Sie! Ihr »68000er«-Poster ist im Abonnementpreis enthalten und gehört Ihnen, auch wenn Sie Ihre Bestellung widerrufen sollten.
FÜR EIN KOSTENLOSES PROBEEXEMPLAR DES »68000er«-MAGAZINS
JA, ich möchte »68000er«, das Magazin der neuen Computer-Generation, kennenlernen.
Senden Sie mir bitte die aktuellste Ausgabe kostenlos als Probeexemplar. Wenn mir »68000er« gefällt und ich es regelmäßig weiterbeziehen möchte, brauche ich nichts zu tun: Ich erhalte es dann regelmäßig frei Haus per Post. Außerdem nutze ich den Abonnement-Preisvorteil von 8 % und bezahle pro Jahr nur 77,– DM statt 84,– DM im Einzelverkauf.
Vorname
Name
Straße
PLZ, Ort
Datum 1. Unterschrift
Mir ist bekannt, daß ich diese Bestellung innerhalb von 8 Tagen bei der Bestelladresse widerrufen kann und bestätige dies durch meine zweite Unterschrift. Zur Wahrung der Frist genügt die rechtzeitige Absendung des Widerrufs.
Datum 2. Unterschrift
Gutschein ausfüllen und absenden an: Markt & Technik Verlag Aktiengesellschaft, Vertrieb, Postfach 1304, 8013 Haar
A 8/9/87

MPS-2000 von Commodore können verschiedene Grafikdichten gewählt werden. Haben Sie als Ausdruckart »grey scale« oder »color« gewählt, so hat die Threshold-Einstellung eine andere Bedeutung als beim Schwarzweiß-Druck. Mit ihr wird die Grafikdichte ausgewählt. Ist der Regler auf »1«, so wird mit einer Dichte von 60 x 60 Punkten pro Zoll ausgedruckt. In der Stellung »10« wird eine Dichte von 180 x 180 Punkten pro Zoll ausgewählt. Die Zwischenwerte stellen die dazwischenliegenden Grafikdichten ein.

Bei anderen Druckertreibern ist es möglich, mit der Einstellung »letter« oder »draft« die Druckqualität des Grafikausdrucks zu verändern. Probieren Sie die Einstellungen einfach mal aus. Beim Programm PageSetter ist es wichtig, den linken und rechten Rand auf ihre Maximalwerte zu stellen. Sonst kann es passieren, daß die durch die Randeinstellungen begrenzte Papierbreite nicht mehr ausreicht.

Vier Farbdrucker für den Amiga

Soviel zur Theorie des Druckens beim Amiga. Wir möchten Ihnen nun vier Drucker vorstellen, die, ebenso wie die im Amiga-Magazin vorgestellten, sehr gut zum Amiga passen. Dabei bestätigt sich wieder, daß zum Amiga eigentlich nur solche Drucker gut passen, die nicht nur Text, sondern auch farbige Grafiken darstellen können.

Ein Alleskönner ist der C.Itoh C 310 CXP (Bild 3), bei dem die Konstrukteure auf besonderen Bedienungskomfort geachtet haben. So entstand ein zu IBM und Epson kompatibler Drukker mit bewährter 9-Nadel-Technik, hoher Druckleistung, NLQ-Schrift und einigen interessanten Details, die wir nun etwas genauer betrachten wollen. Im Gegensatz zu den meisten Druckern hat der C 310 CXP überhaupt keine DIL-Schalter, sondern bedient sich eines völlig anderen Verfahrens. Mit den vier Bedientasten auf der Gehäusevorderseite (Online, Formfeed, Linefeed und NLQ) können insgesamt bis zu 21 verschiedene Einstellungen vorgenommen werden. Diese 21 »programmierbaren DIL-Schalter« speichern die gewünschten Einstellungen in einem CMOS-RAM, dessen Inhalt durch eine Batterie auch nach dem Ausschalten erhalten bleibt. Auch bei der Frage der Traktorart hat man sich einiges überlegt. Beim C 310 CXP sind die Konstrukteure auf einen besonders raffinierten Trick gekommen, denn der C 310 CXP hat sowohl einen Schubtraktor als auch einen Zugtraktor. Beide sind in einem einzigen Traktor vereinigt, der entweder hochgeklappt (Zugtraktor) oder heruntergeklappt (Schubtraktor) werden kann. Dieses Prinzip verdient allergrößtes Lob und sollte von allen Druckerherstellern übernommen werden. Dabei kann das Papier nicht nur wie üblich von hinten zugeführt werden, sondern auch von unten, denn unter dem Drucker befindet sich ein Einlaßschlitz für Endlospapier. Damit hat man es aber noch nicht bewenden lassen, denn auf der linken Gehäuseseite befindet sich eine Einschubhalterung für Schriftmodule verschiedenster Art.



Bild 3. Der C 310 CXP ist ein Drucker mit Liebe zum Detail

Bisher werden die Module OCR-B, Italic und Letter Gothic angeboten (je 98 Mark). Die Schriftmodule werden einfach eingesteckt und schon hat man eine weitere NLQ-Schrift zur Verfügung.

Grundsolider Aufbau

Neben diesen sehr nützlichen und wichtigen Zusatzeinrichtungen soll nicht unerwähnt bleiben, daß der C 310 CXP extrem solide und stabil aufgebaut ist. Das Grundgerüst der Druckers besteht aus massivem Stahl, der Druckkopf gleitet auf dicken Stahlrohren und das gesamte Gehäuse ist innen mit Schaumstoff ausgepolstert. Das bleibt natürlich nicht ohne Wirkung auf die Geräuschentwicklung beim Drucken. Der C 310 CXP ist der leiseste bisher getestete Nadel-Matrixdrucker, bei dem nicht an der Leistung (Tabelle 2) und am Schriftbild (Bild 4) gespart wurde. Nun wäre der C 310 CXP mit den beschriebenen Leistungen einer der besten Drucker, die wir je getestet haben, doch die Konstrukteure geben noch ein besonderes Bonbon dazu. Der C 310 CXP ist nämlich farbfähig und das auch noch serienmäßig. Durch einfaches Austauschen des Farbbandes erhält man die Farbfähigkeit eines JX-80 hinzu, wie gesagt — ohne Aufpreis (Bild 5).

C.Itoh C 310 CXP
NLQ-Schrift
NLQ-Kursiv
Normalschrift
Kursivschrift
Elite-Schrift
Schmalschrift
Breit
Fettdruck
Doppeldruck
Hoch- und tief

Bild 4. Gutes Schriftbild braucht nicht unbedingt 24 Nadeln — C 310 CXP

Mit einem Listenpreis von 1998 Mark ist der C 310 CXP sicherlich nicht gerade billig. Aber gerade durch sein durchdachtes Gesamtkonzept, zu dem auch die sinnvolle Funktion gehört, den Papiertransport für das Endlospapier abzuschalten, wenn Einzelblätter verwendet werden, ist der C 310 CXP für den Amiga bestens geeignet.

Die Workbench 1.2 bietet endlich eine große Auswahl an Druckern. Dazu gehört auch der Oki ML 292 (Bild 6), der als Drucker mit 18 Nadeln am Amiga ja nicht ohne weiteres zum Grafikdruck bewegt werden kann. Stellt man aber in den Preferences den ML 292 ein, so braucht man sich keine weiteren Gedanken darüber zu machen, ob nun neun oder 18 Nadeln verwendet werden.

Farbe mit 18 Nadeln

Wichtig ist dabei nur, daß man den ML 292 mit IBM-Schnittstellenmodul verwendet, denn nur dann werden die Steuerbefehle für den Grafikdruck (Bild 7) richtig verstanden. Da die Farbfähigkeit beim ML 292 fest eingebaut ist, braucht man übrigens nur zwischen einem mehrfarbigen und einem schwarzen Farbband zu wechseln um den entsprechenden Modus zu aktivieren. Aber nicht nur bei der Grafik (farbig oder Schwarzweiß) zeigt der sehr solide aufgebaute ML 292 seine Qualitäten. Auch beim Textdruck (Bild 8) machen sich die 18 Nadeln erfreulich bemerkbar. Die Schrift ist sehr schön und wird trotzdem relativ schnell mit 100 Zeichen/Sekunde zu Papier gebracht. In der EDV-Schrift erreicht der ML 292 sogar 200 Zeichen/Sekunde. Die Handhabung des Druckers ist ebenso problemlos wie das Einspannen des Papiers. Dabei ist der Traktor direkt links und rechts neben der Schreibwalze angebracht. Das hat zwar den Nachteil, daß man den Traktor kaum verstellen kann, dafür spart man aber eine ganze Menge Papier. Wer will kann das Papier aber auch von unten zuführen, was bei Verwendung eines Druckerständers zu enormer Platzersparnis führt.

Die technischen Daten (Tabelle 2) sowie der Eindruck, den der ML 292 während unseres Tests hinterlassen hat, be-



Bild 5. Beispiel einer farbigen Hardcopy mit dem C 310 CXP

Modellbezeichnung	Microline ML 292 Okidata GmbH Hansa-Allee 187 4000 Düsseldorf 11	C 310 CXP C. Itoh GmbH Roßstr. 96 4000 Düsseldorf 30	TPX 80 C. Itoh GmbH Roßstr. 96 4000 Düsseldorf 30	Okimate 20 Okidata GmbH Hansa-Allee 187 4000 Düsseldorf 11
Empfohlener Preis:	1515 Mark + 420 Mark f. Interface	1998 Mark	1098 Mark	888 Mark
Abmessungen (B x H x T):	367 x 105 x 305 mm	438 x 122 x 330 mm	380 x 80 x 290 mm	330 x 60 x 190 mm
Farbband Preis Farbe: S/W:	48 Mark 32 Mark	55 Mark 30 Mark	22 Mark 18 Mark	17 Mark 14 Mark
Druckkopf:	18 Nadeln	9 Nadeln	24 Thermoelemente	24 Thermoelemente
Gewicht:	5,7 kg	9,6 kg	5,5 kg	2,8 kg
Zeichenmatrix (H x B):	9 x 9	9 x 9	12 x 15	7 x 14
NLQ-Matrix (H x B):	17 x 17	27 x 17	24 x 15	14 x 14
Papiersorten: einzel endlos	bis 254 mm 76 mm — 254 mm	bis 279,4 mm 107,9 mm — 279,4 mm	89 mm — 257 mm 89 mm — 257 mm	127 mm — 254 mm 127 mm — 254 mm
Zeichensätze:	IBM, ASCII	IBM, ASCII	IBM, ASCII	IBM, ASCII
Zeichen/Zeile:	137	137	136	132
Durchschläge:	2	2	keine	keine
Hexdump:	Ja	Ja	Ja	Nein
Selbsttest:	Ja	Ja	Ja	Ja
Pufferspeicher:	15 KByte	10 KByte	2 KByte	keine Angabe
Halbautom. Einzelblatteinzug:	Ja	Ja	Nein	Nein
Geschwindigkeit EDV: angegeben: gemessen:	200 Zeichen/s	250 Zeichen/s	80 Zeichen/s 32 Zeichen/s	80 Zeichen/s 32 Zeichen/s
Geschwindigkeit NLQ: angegeben: gemessen:	100 Zeichen/s 99 Zeichen/s	33 Zeichen/s 36 Zeichen/s	45 Zeichen/s 24 Zeichen/s	40 Zeichen/s 24 Zeichen/s
Probetext EDV: Probetext NLQ:	1:16 Minuten 1:59 Minuten	1:23 Minuten 6:15 Minuten	8:19 Minuten 11:11 Minuten	8:40 Minuten 11:50 Minuten
Ladbarer Zeichensatz:	Ja	Ja	Ja	Ja
MTBF (in Stunden):	4000	keine Angaben	keine Angaben	keine Angaben
Lebensdauer des Druckkopfes:	200 Millionen Zeichen	100 Millionen Zeichen	keine Angaben	keine Angaben
Geräuscheindruck:	durchschnittlich laut	durchschnittlich leise	sehr leise	sehr leise
Grafikmodi:	480, 576, 640, 720, 960, 1920	480, 576, 690, 720, 960, 1920	480, 576, 640, 720, 960, 1920	480, 960, 1920, 960 mit 24 Elementen
Schriftarten:	Pica, Elite, proportional, schmal, breit, hoch, tief, unterstrichen, Italic	Pica, Elite, proportional, schmal, breit, hoch, tief, unterstrichen, Italic	Pica, Elite, proportional, schmal, breit, hoch, tief, unterstrichen, Italic	Pica, Elite, schmal, breit, hoch, tief, unterstrichen, Italic
Funktionstasten:	Online, Linefeed, Formfeed, Topot Form	Online, Linefeed, Formfeed, Topot Form, NLQ	Online, Formfeed, NLQ, Druckgeschwindigkeit, Intensität	Intensität, Online
Besondere Funktionen:	Farbe fest eingebaut	Farbe fest eingebaut	IBM und ESC/P-kompatibel	IBM-kompatibel
Sonderzubehör:	Einzelblatteinzug	Einzelblatteinzug	—	—
Handbuch: Beispiele: Note für Handbuch:	2 x deutsch MS-Basic 2	deutsch MS-Basic 2	englisch + deutsch keine 4—5	deutsch MS-Basic 2

Tabelle 2. Technische Daten und Leistung der vier Testdrucker im Vergleich

OKI ML 292
NLQ-Schrift
NLQ-Kursiv
Normalschrift
Kursivschrift
Elite-Schrift
Schmalschrift
Breit
Fettdruck
Doppeldruck
Hoch- und tief

Bild 8. Die Schriftqualität des ML 292 mit 18 Nadeln ausgedruckt — fast perfekt

stätigen, daß der ML 292 mit 1935 Mark zwar nicht billig, aber gemessen an seiner Leistung preiswert ist.

Klein, preiswert (888 Mark) und farbig zeigt sich der Okimate 20 (Bild 9), ein Thermo-Transfer-Drucker, den es passend zum Amiga mit einer Centronics-Schnittstelle gibt.

Farbe aufs Papier gebrannt

Auf der Workbench 1.2 findet sich ein passender Druckertreiber, so daß die Installation kein Problem darstellt. Der Okimate 20 besteht zu mindestens 80 Prozent aus Plastik und unhandlichen Hebeln. So gestaltet sich das Arbeiten mit diesem Winzling mitunter auch schon etwas umständlich. Angefangen bei der Farbbandkassette (Schwarzweiß oder farbig), über das Einlegen des Papiers (nur Einzelblätter und Rollenpapier), bis zum Programmieren des Druckers fordert der Okimate 20 einiges an Gewöhnung und Geduld. Trotzdem ist das, was der Okimate 20 auf das Papier zaubert (Bild 10) fast makellos; die Farben leuchten und werden fast korrekt wiedergegeben, nur

Bild 6. Der ML 292 läßt sich direkt im Preferences Tool installieren

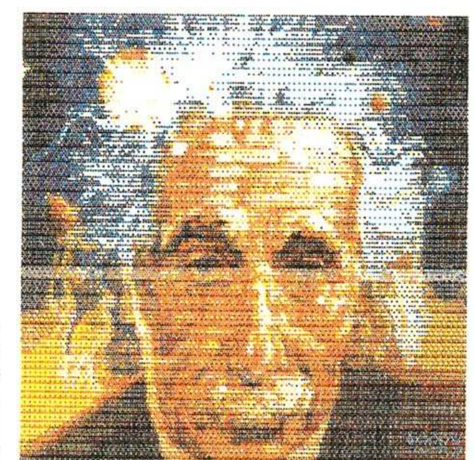


Bild 7. Farbige Hardcopies funktionieren mit dem ML 292 problemlos



Bild 9. Klein, leicht und farbig — der Okimate 20. Der Farbdruck ist für ihn ebenso wie der Textdruck kein Problem.

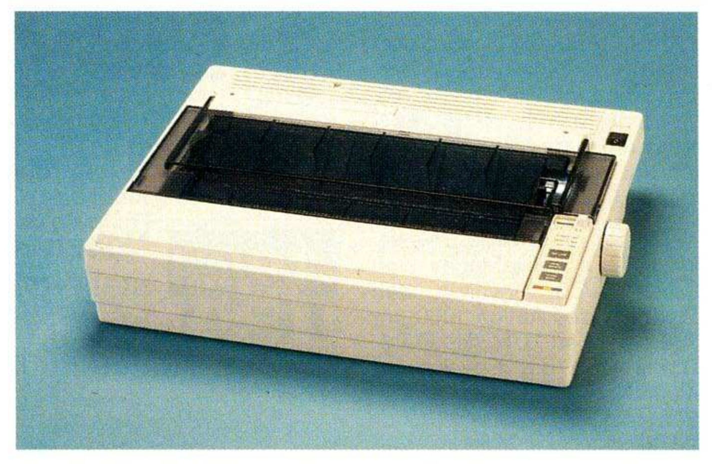


Bild 12. Der TPX 80 gefällt durch gutes Design und hohe Leistungsfähigkeit. Außerdem ist er leicht zu bedienen.

zwischen den Grafikzeilen bleibt manchmal ein störender Spalt frei. Für den Okimate gibt es zwar auch ein schwarzes Farbband, für den Textdruck ist er aber durch seine relativ niedrige Geschwindigkeit (effektiv unter 80 Zeichen/Sekunde) und das teure Farbband (14 Mark Schwarzweiß), trotz der guten NLQ-Schrift (Bild 11), nur bedingt geeignet. Die Domäne des Okimate 20 ist hauptsächlich der farbige Druck, vornehmlich in Form von Bildschirm-Hardcopies. So eine Hardcopy braucht dann allerdings seine Zeit und kostet auch einiges, denn aus dem 17 Mark teuren farbigen Farbband kann man nicht mehr als fünf bis acht Hardcopies anfertigen. Mit seinem Preis von 888 Mark ist der Okimate 20 deshalb auch eher eine sinnvolle Ergänzung zu einem guten Nadeldrucker (Tabelle 2).

Der C.Itoh TPX 80 (Bild 12) arbeitet nach einem ähnlichen Prinzip wie der Okimate 20, zeichnet sich aber durch eine wesentlich höhere mechanische Stabilität und mehr Bedienungskomfort aus. Der TPX 80 besitzt außerdem neben seiner Farbfähigkeit (Bild 13) auch einen vollwertigen NLQ-Textmodus (24 x 15 Punkte). Extrem leise, mit gestochen scharfem Schriftbild (Bild 14) und sogar relativ flott (80 Zeichen/Sekunde normal und 45 Zeichen/Sekunde in NLQ) brennt der TPX 80 die Farbe (Farbband 22 Mark, Schwarzweiß 18 Mark) auf das Papier. Den Anspruch, nicht nur ein Grafikdrucker zu sein, unterstreicht der TPX 80 zusätzlich durch seinen umfangreichen Befehlssatz, der nach ESC/P-Norm, also Epson-kompatibel (Tabelle 2) genormt ist. Damit ist der TPX 80 mit dem Amiga problemlos anzusteuern, man installiert ihn zum Beispiel als Epson JX-80 im Preferences-Tool. Zum Preis von 1098 Mark erhält man im TPX 80 einen sehr soliden Drucker mit excellenten Text- und Grafikfähigkeiten, der außer den bauartbedingten Problemen eines Thermo-Transferdruckers kaum Nachteile aufweist.



Bild 10. Durch Übereinanderdrucken erzeugt der Okimate 20 die Mischfarben

OKIMATE 20
NLQ-Schrift
NLQ-Kursiv
Normalschrift
Kursivschrift
Elite-Schrift
Schmalschrift
Breit
Fettdruck
Doppeldruck
Hoch- und tief

Bild 11. Schriftprobe des Okimate 20 auf Spezialpapier

Der Umgang mit Druckern und die Wiedergabe der herrlichen Grafiken des Amigas gehören zu einer der schönsten Aufgaben, die man mit einem Computer machen kann.

Drucken am Amiga macht Spaß

Stimmt die Ausrüstung und die Installation, so ist der Umgang mit einem Drucker beim Amiga sehr einfach. Probleme, wie man sie von anderen Computern, wie zum Beispiel dem C 64 kennt, fallen einfach unter den Tisch. Das liegt vor allem daran, daß der Amiga mit allen Schnittstellen ausgerüstet ist, die ein moderner Computer eben so braucht. Dadurch entfällt zum Beispiel das leidige Interface-Problem, bei dem nicht nur der Drucker, sondern auch das Interface falsch eingestellt sein kann. Auch die Anpassung an alle noch kommenden Druckergenerationen einschließlich der Laserdrucker ist sichergestellt, denn man braucht nicht gleich das gesamte Betriebssystem ändern, sondern lediglich den Druckertreiber. Dies alles macht Drucken so interessant, besonders dann, wenn man einen Farbdrucker besitzt, denn es kann nur nochmals betont werden, daß ein einfarbiger Drucker am Amiga vieles von den Fähigkeiten des Amiga verschenkt. Sei es, daß man gerne für private oder geschäftliche Zwecke farbige Hardcopies von beispielsweise grafischen Werteausgaben oder Diagrammen anfertigen möchte. Denkbar wären aber auch Ausdrucke von besonders gelungenen Bildern, die zur Auflockerung von Briefen verwendet werden können. Oder aber auch »nur«, um sich ein besonders gelungenes Bild an die Wand hängen zu können. Für welche der vorgestellten Techniken man sich letztendlich entscheidet, ist sicherlich eine Frage der persönlichen Interessen und des zur Verfügung stehenden Budgets.

(Gerald Höfer/aw)



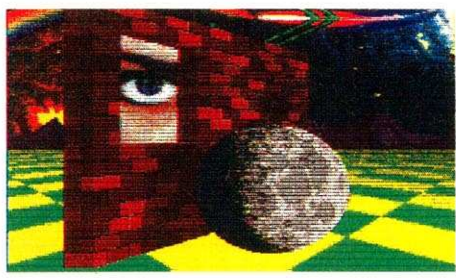


Bild 13. Der TPX 80 druckt die Zeilen genauer untereinander als der Okimate 20

C.Itoh TXP 80
NLQ-Schrift
NLQ-Kursiv
Normalschrift
Kursivschrift
Elite-Schrift
Schmalschrift
Breit
Fettdruck
Doppeldruck
Hoch- und tief

Bild 14. Der TPX druckt auf Spezialpapier fast perfekt

C.Itoh, Roßstr. 96, 4000 Düsseldorf 30
Okidata, Hansa-Allee 187, 4000 Düsseldorf 11

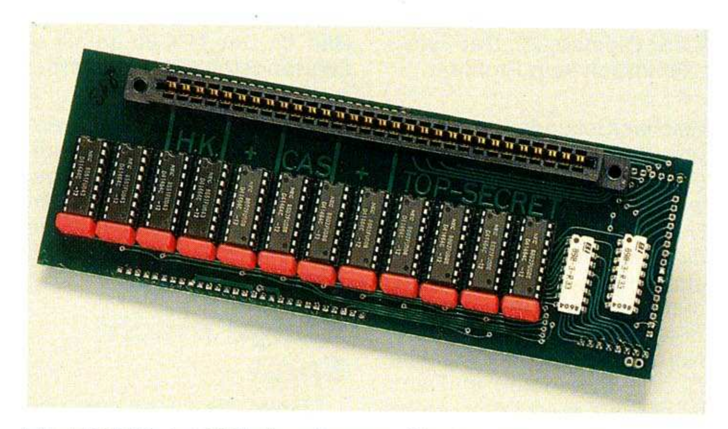
1 MByte RAM für den Amiga 1000

AMIGA TEST

Neben einigen anderen Herstellern bietet jetzt auch C.A.S. eine 768-KByte-Erweiterung für den Amiga 1000 an, die anstelle der 256-KByte-Erweiterung eingesteckt wird.

Die neue RAM-Erweiterung »RaMega« von C.A.S. erweitert den Speicher des Amiga 1000 um 768 KByte auf 1 MByte RAM. Dabei wird die Erweiterung nicht am Expansionsbus, sondern in der Einsparung an der Vorderseite des Amiga 1000 eingesteckt.

Das Problem bei dieser Art von RAM-Erweiterungen besteht darin, daß die Daten- und Adreßleitungen am vorderen Erweiterungsport nur bis 512 KByte ausgelegt sind. Ohne den Amiga aufzumachen und einige Drähte anzulöten, geht es also in keinem Fall. Der Grund, trotzdem solche RAM-Erweiterungen zu bauen, besteht darin, daß kein zusätzlicher, aufwendiger Controller nötig ist, um das RAM zu verwalten. Dies führt zu erheblichen Preiseinsparungen. Auf der Karte werden 24 RAM-Bausteine vom Typ 41464 mit 120 ns Zugriffszeit verwendet, die auf zwei Platinen verteilt sind. Diese Platinen sind an ihren Lötseiten zusammengeklebt (Sandwichtechnik). Das führt allerdings dazu, daß die RAM-Erweiterung zu »dick« wird, um noch unter die normale Abdeckung zu passen.



Die 768-KByte-RAM-Erweiterung für den Amiga 1000 von C.A.S. zum Einbau in die Front-Aussparung

Endlich genügend Speicher

Aus diesem Grund wird auch eine neue Frontklappe mitgeliefert (in Amiga-grau oder Plexiglas), die etwas größer als die normale ist und etwa 1,5 cm herausragt. Es können aber hierbei bei manchen (älteren) Amiga Probleme entstehen, wenn man die Tastatur noch unter den Rechner schieben möchte, da die Abdeckung auch etwas nach unten herausragt. Abhilfe schafft aber das Unterlegen von beispielsweise einem dünnen Buch, das den Amiga 1000 vorne etwas anhebt. Schon paßt die Tastatur wieder unter den Computer.

Bei der C.A.S.-RaMega-Erweiterung müssen insgesamt 12 Drähte verlötet und vier IC-Beinchen im Innern des Amiga durchgetrennt werden, um in den Genuß des zusätzlichen Speichers zu kommen. Vorteilhafter für den Anwender ist es in diesem Fall aber, nicht die IC-Beinchen durchzutrennen, sondern die betreffenden Bausteine zu sockeln. Dies ist zwar mit etwas Mehraufwand verbunden, hilft aber, größere Schäden von vornherein zu vermeiden. Der Einbau geht relativ einfach vonstatten und ist in einer guten halben Stunde zu bewerkstelligen. Die mitgelieferte Bauanleitung glänzt zwar nicht gerade durch Ausführlichkeit, erfüllt aber ihren Zweck dennoch. Traut man sich den Einbau selbst nicht zu, so übernimmt C.A.S. dies für einen Aufpreis von 98 Mark.

Die Garantie für den Computer geht natürlich in jedem Fall verloren. Man sollte sich also genau überlegen, ob man vielleicht doch noch vor hat, seinen Amiga während der Garantiezeit zu verändern. Es ergeben sich übrigens keine Probleme, ob die RAM-Erweiterung in einen Amiga mit oder ohne Huckepackplatine eingebaut wird; es funktioniert beides. Im Lieferumfang ist neben einigen RAM-Test-Programmen ein Kickstartgenerator zu finden.

Er macht aus jeder gewöhnlichen Kickstart-Version 1.2 eine Version »1.2+«. Man sollte immer mit dieser arbeiten, um eine korrekte Funktion der RaMega-Karte zu garantieren.

Kickstart-Änderungen nötig

In ihr werden nämlich die 1024 KByte (768 KByte Erweiterung plus den auf der Mutterplatine vorhandenen 256 KByte) softwaremäßig auf Chip-RAM (das sind die untersten 512 KByte, auf die die Spezialchips des Amiga zugreifen können) und Fast-RAM verteilt, was bei am Expansionsbus angeschlossenen RAM-Erweiterungen nicht nötig ist. Benutzt man die normale Kickstart-Version weiter, so erkennt der Computer den ganzen Speicher als Chip-Memory an, was zu Komplikationen führen kann, da theoretisch nicht mehr als 512 KByte Chip-Memory angesprochen werden können. Wird aber mit »Kickstart 1.2+« gearbeitet, kommen diese Probleme nicht vor und der Speicher wird korrekt verwaltet (512 KByte Chip-RAM und 512 KByte Fast-RAM). Allerdings ist das hier »emulierte« Fast-RAM nicht ganz so schnell wie »echtes« Fast-RAM, sondern genauso schnell beziehungsweise langsam wie normales Chip-Memory. Dies macht sich aber nicht sonderlich bemerkbar.

Ein echter Clou liegt darin, ohne irgendwelche weiteren Programme die zusätzlichen 512 KByte softwaremäßig (!) ganz abschalten zu können (so daß nur noch 512 KByte Chip-Memory erkannt werden). Dies geschieht, indem man während des Reset mit <CTRL Amiga Amiga> beziehungsweise <CTRL Commodore Amiga> die linke Maustaste drückt (nein, nicht gleichzeitig, dann bräuchte man ja drei Hände, sondern kurz, nachdem der Reset ausgelöst wurde). Jetzt ist die RAM-Erweiterung vollständig abgeschaltet. Nach nochmaligem Reset ist wieder der volle Speicher von 1 MByte vorhanden. Mit der C.A.S.-Erweiterung funktionierten alle ausgetesteten Programme hervorragend und ohne Komplikationen (bis auf die Programme, die sowieso nicht mit mehr als 512 KByte laufen, wie etwa Deluxe-Video 1.0). Nur die Recoverable RAMDisk-Software von ASDG machte Schwierigkeiten und war nicht dazu zu bewegen, ein Programm ins RAM zu übernehmen. Nach Aussage von C.A.S. wird aber bald ein kostenloses Utility-Programm erscheinen, das diesen Mangel beheben soll.

Erweiterung zu empfehlen?

Die RAM-Erweiterung ist interessant für jeden Amiga 1000-Besitzer, der nur wenig Geld hat und den es nicht stört, daß die Erweiterung eine etwas größere Frontabdeckung erforderlich macht. Die RAM-Erweiterung verhält sich zu vorhandener Software sehr kompatibel. Leider ist der Einbau nicht von jedem zu bewerkstelligen, da Lötarbeiten erforderlich sind. Diese kleinen Mankos sollten aber nicht stören, da man als Endprodukt einen Amiga 1000 mit 1 MByte Speicher vor sich hat.

(Ottmar Röhrig/dm)

Info: C.A.S., Sprendlinger Landstr. 71, 6050 Offenbach, Tel. 069/84 20 13. Preis: 498 Mark (inkl. MwSt.) mit Garantie und Update-Service

Aus RGB wird PAL

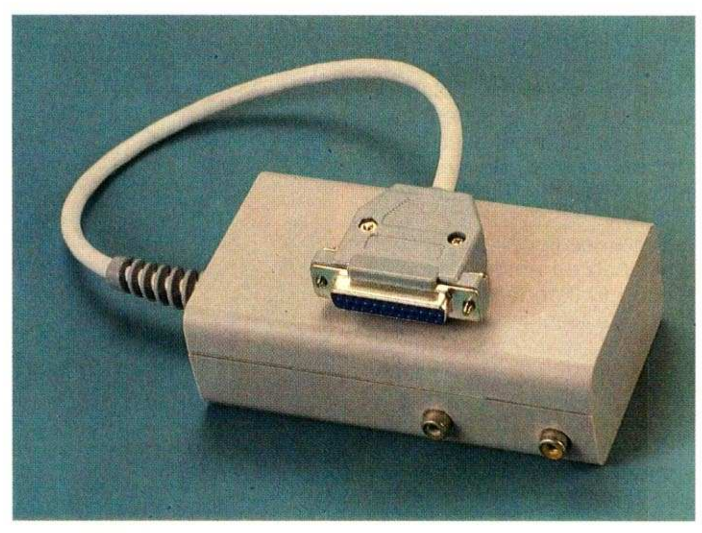
AMIGA TEST

Das C.A.S.-PAL-Set ermöglicht den Anschluß aller Amigas an Fernsehgeräte oder Recorder mit Video- oder HF-Eingang.

Nicht jeder kann es sich leisten, gleichzeitig mit seinem Amiga einen Monitor zu kaufen. Hier muß erst einmal der heimische Fernseher herhalten. Bei neueren Amiga 1000-Modellen (PAL-Amigas) und Fernsehern ist das kein Problem, da beide einen Videoausgang/-eingang besitzen und lediglich ein Kabel zur Verbindung genügt. Hat man aber einen Fernseher, der keinen Video-, sondern nur einen HF-Antenneneingang besitzt, wird es schon komplizierter. Hier muß aus dem Videosignal ein HF-Signal moduliert werden, um das Bild genießen zu können. Sollte aber der Amiga älteren Datums sein, muß aus der (amerikanischen) NTSC-Norm am Videoausgang noch die (deutsche) PAL-Norm »gemacht« werden. Alle diese Aufgaben übernimmt das PAL-Set von C.A.S (Bild 1). Es wird am RGB-(Monitor-)ausgang des Amiga angeschlossen und stellt an zwei Chinch-Buchsen je ein PAL-Video- sowie ein PAL-HF-Signal zur Verfügung. Damit soll dann ein Fernseher oder Videorecorder über die »Video-In«-Buchse anschließbar sein. Das HF-Signal wird benötigt, um einen Fernseher über die normale Antennenbuchse anzuschließen. Dazu muß allerdings zuerst per Hand eine Abstimmung der Frequenzen von PAL-Set und Fernseher erfolgen. Das an den Ausgängen anliegende Bild ist natürlich auf keinen Fall mit der Brillanz des RGB-Signals, wie Monitore es darstellen, zu vergleichen. Die Farben sind etwas blasser, die Konturen nicht so scharf und das Bild flimmert leicht; aber bis das Geld für einen RGB-Monitor vorhanden ist, kann man recht gut damit arbeiten. Ein anderes Anwendungsgebiet liegt in der Herstellung von kleinen Filmen, die sich dann auf Videorecordern aufzeichnen lassen. So können auch Titelvorspänne zu eigenen Filmen fast professionelles Aussehen gewinnen. Da sich das C.A.S-Modul auch am Amiga 2000 verwenden läßt, wird laut Aussagen des Herstellers die A2000-Video-Steckkarte im wesentlichen ersetzt. Das PAL-Set gibt es in drei Ausführungen: Zum einen die oben beschriebene, mit Video- und HF-Ausgang zum Preis von 198 Mark. Zum anderen gibt es Versionen mit nur einem PAL-Video-Ausgang (Preis 139 Mark) oder nur mit HF-Ausgang (159 Mark).

(Ottmar Röhrig/jk)

C.A.S., Sprendlinger Landstr. 71, 6050 Offenbach, Tel. 069/842013



Das C.A.S.-PAL-Set für die Amiga-Familie

Die Trickkiste

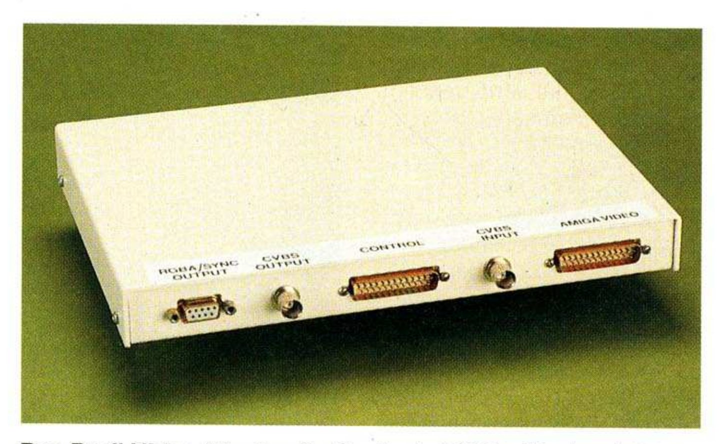
AMIGA TEST **Dachten Sie schon einmal daran, Computergrafik mit Bildern von einem Videorecorder oder einer Kamera zu mischen, um Urlaubsfilme oder Präsentationen reizvoller zu gestalten? Dann könnte das Genlock-Interface »A8600« ein nützliches grafisches Hilfsmittel für Sie werden.**

Wenn der eigene Videofilm mit Trickfilmvorspann und eingeblendetem Lauftext Beifallsstürme im Heimkino hervorruft, der Manager die Umsatzbilanz mit animierter Tortengrafik vor dem Produktfilm seinen Aktionären präsentiert, im Schaufenster der Videomonitor mit Werbefilm und Text zum Kauf anregt, war sicherlich das Grafikwunder Amiga und ein Genlock-Interface am Werk.

Mit Hilfe des Genlock-Interface können zwei Videosignale miteinander gemischt werden. Dies kann nicht so einfach geschehen, wie bei einem Tonsignal, bei dem bekanntlich die verschiedenen Signale nur miteinander verbunden werden.

Ein Videosignal enthält noch eine Vielzahl anderer Signale für die Elektronenstrahlsteuerung des Monitors.

Die Entwickler des Amiga haben jedoch vorausschauend geplant und ihrer Grafikmaschine einen externen Synchronisationseingang spendiert. Jetzt kann das Videosignal des Amiga mit einem externen Videosignal »ferngesteuert« werden. Diese Aufgabe übernimmt das Genlock-Interface. Jedoch kann man mit einem Genlock-Interface keine Bilder digitalisieren oder in den Computer einlesen.



Das Profi-Video-Mischpult »Genlock A8600« für den A1000

Das Genlock-Interface A8600

Dieses englische, auf deutsche (PAL)-Videoverhältnisse umgebaute Gerät besitzt keinerlei Regler oder Knöpfe. Die gesamte Steuerung erfolgt über den Parallelport des Amiga 1000. Das klingt anfänglich sehr kompliziert, ist aber durch die auf Diskette mitgelieferte, leistungsfähige Software sehr einfach zu bewerkstelligen. Auf der Diskette sind alle nötigen Programme einschließlich einer »Genlock.library« für den Programmierer sowie ein kleines Demonstrationsvideo enthalten. Ein Installationsprogramm kopiert automatisch die Genlock-Steuerung auf jede Workbench oder auf Programmdisketten, wie beispielsweise »Deluxe-Video« oder »Deluxe-Paint«. So kann ein Programm wie »Deluxe-Video« im Filmscript alle Steuersignale enthalten und damit die Hardware steuern. Die Funktionen des Genlock-Interfaces sind aber auch durch ein Fenster-Menü mit der Maus steuerbar. Dieses Fenster erscheint auch auf dem Bildschirm und ist damit bei der Aufnahme auf dem Videorecorder sichtbar. Es sei denn, kurz vor Betätigen der Aufnahmetaste wird ein zweiter (Effekt)-Screen eröffnet und der (Arbeits)-Screen geschlossen.

Das in einem stabilen grauen Stahlblechgehäuse befindliche Gerät paßt vorbildlich unter den Amiga 1000. Dabei können aber die drei RGB-Cinchbuchsen auf der Vorderseite des Gerätes nicht mehr verwendet werden, da sie vom Sockel des Amiga verdeckt werden. Die mitgelieferten Anschlußkabel sind so kurz geraten, daß eine Verbindung mit dem Amiga 500 und Amiga 2000 unmöglich ist. Anstatt des eigentlich passenden 23poligen D-Sub-Steckers wurde ein »beschnittener« 25poliger D-Sub-Stecker verwendet. Auch in England scheint es für diesen commodorespezifischen Stecker Beschaffungsschwierigkeiten zu geben. Die Anschlußkabel werden dem sonst sehr professionell gebautem Gerät nicht gerecht. Auch wurden bei dem sowieso schon sehr wackeligen Amiga-Monitorstecker die nicht benutzten Kontakte einfach nicht eingesetzt. Wird der Monitor auf einem drehbaren Monitorständer betrieben, fällt der Stecker ständig aus der Monitorbuchse.

Im Inneren des Gerätes befinden sich zwei sauber aufgebaute doppelseitig-kaschierte, dichtbepackte Platinen. Leider wurden alle Bezeichnungen von den Bauteilen entfernt und auch kein Schaltplan mitgeliefert. Im Garantiefall ist man also auf den Service des Händlers angewiesen.

Durch die Anpassung des in England gebauten Genlock-Interfaces an die deutsche Fernsehnorm befinden sich einige »fliegend« verdrahtete Bauteile im Gerät. Hier wäre eine solidere Lösung wünschenswert.

Nach Anschluß des A8600 an den Amiga wird die CVBS-Input-Buchse mit dem Cinch-Ausgang des Wiedergaberecorders oder Bildplattenspielers angeschlossen (hierfür gibt das Handbuch ausführliche Hinweise). Die CVBS-Output-Buchse wird mit der Eingangsbuchse des Aufnahmerecorders verbunden.

Für den Philips-Bildplattenspieler besteht die Möglichkeit, ihn ebenfalls vom Genlock-Interface aus zu steuern.

Auf der mitgelieferten Programmdiskette befindet sich ein kleines und eindrucksvolles Demonstrationsprogramm. Der eigene Videofilm wird damit in den Zeichentrickfilm eingebunden. Obwohl die Bildqualität einige sichtbare Verluste erleidet, kann man das Ergebnis als absolut zufriedenstellend bezeichnen. Vor allem, da man sich den unvergleichlich höheren (finanziellen) Aufwand solcher Effektgeräte in professionellen Fernsehstudios vor Augen halten muß. Das A8600 liefert das beste gemischte Bild aller uns derzeit bekannten, für den Amiga erhältlichen Geräte.

Mit vier softwaremäßig einstellbaren »Schaltern« lassen sich die zu bearbeitenden maximal acht Filterfarben einstellen. Es handelt sich jedoch nicht um einzelne genau definierte Farben, sondern um Farbbereiche. Dies resultiert aus Bauteiletoleranzen und prinzipiell aus der analogen Steuerung des Genlock-Interfaces. Möchte man beispielsweise alle Weißtöne ausblenden, so werden auch hellgelbe Farbtöne miterfaßt.

Mit Hilfe weiterer Softwareschalter wählt man zwischen dem Amiga-Bild als Vordergrund und einem Videobild als Hintergrund, oder umgekehrt.

Der Videofilmer kann jede nur erdenkliche Trickfilmblende mit Deluxe-Paint erstellen und mit dem automatischen Fade-In-/Fade-Out-Programm ein- und ausblenden.

Fazit A8600

Die etwa 20 Seiten starke englische Anleitung gibt leider nur wenige Hinweise und Erklärungen. Bei einem Gerät in der Preisklasse eines Computers (Preis A8600: etwa 1300 Mark) wäre eine umfangreichere deutsche Anleitung sicher kein Luxus.

Das Gerät besitzt auch keine postalische FTZ-Zulassung.

Sieht man aber über diese Punkte hinweg, steht dem Hobbyanwender und Videofreak mit dem Genlock-Interface A8600 ein qualitativ hochwertiges, gut durchdachtes Videomischpult zur Verfügung. Durch die Software wird dem Videofilmer ein äußerst anwenderfreundliches, wenn auch nicht ganz billiges Genlock-Interface zur Verfügung gestellt, dessen Kauf sich bei genügender Nutzung lohnen dürfte. (Joschy Polierer/dm)

Interplan, Nymphenburgerstr. 134, 8000 München 19, Tel. 089/1234066
Merkens EDV, Fuchstanzstr. 6a, 6231 Schwalbach, Tel. 06196/3026
Video Loft Film Hard & Soft, Fiedlerstr. 22-32, 3500 Kassel, Tel. 0561/877928

Die Amiga-Bibliothek

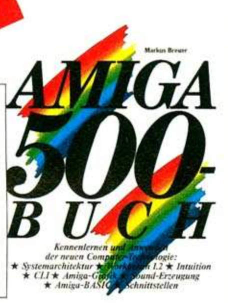




H.-R. Henning
Programmieren mit Amiga-Basic
1987, 360 Seiten, inkl. Diskette
Eine praxisbezogene Einführung in die Programmierung mit Amiga-Basic. Mit 100 Programmen und vielen Beispielen sowie einem Malprogramm und einer leistungsfähigen Dateiverwaltung.
Bestell-Nr. 90434
ISBN 3-89090-434-3
DM 59,–/sFr 54,30/öS 460,20

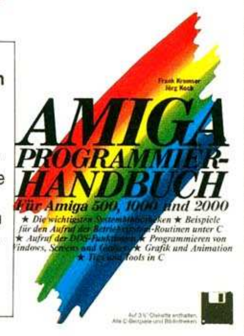
M. Breuer
DELUXE Grafik mit dem Amiga
1987, 370 Seiten.
Schrittweise Einführung anhand überschaubarer Beispiele und Anwendung der wichtigsten Befehle. Datenaustausch zwischen den Programmen. Tips und Tricks für jeden Anwender.
Best.-Nr. 90412
ISBN 3-89090-412-2
DM 49,–/sFr 45,10/öS 382,20





M. Breuer
Das Amiga 500-Buch
1987, ca. 450 Seiten
Eine ausführliche Einführung in die Bedienung des Amiga 500. Das Handbuch dient als Nachschlagewerk beim alltäglichen Einsatz.
Bestell-Nr. 90522
ISBN 3-89090-522-6
DM 49,–/sFr 45,10/öS 382,20





Kremser/Koch
Amiga Programmierhandbuch
1987, 390 Seiten, inkl. Diskette
Eine Super-Einführung in die »Interna« des Amiga: die wichtigsten Systembibliotheken, die das Betriebssystem zur Verfügung stellt, werden ausführlich anhand von Beispielen erklärt.
Bestell-Nr. 90491
ISBN 3-89090-491-2
DM 69,–/sFr 63,50/öS 538,20



M. Kohlen
Grafik auf dem Amiga
1987, 337 Seiten
Kennenlernen der fantastischen Möglichkeiten des Amiga-Computers. Ein Programmierkurs für Anfänger und Fortgeschrittene mit vielen Beispielen.
Best.-Nr. 90236
ISBN 3-89090-236-7
DM 49,-/sFr 45,10/öS 382,20



Bantam Books
Das Amiga-DOS-Handbuch für Amiga 500, 1000 und 2000
1987, ca. 300 Seiten
Die Pflichtlektüre für jeden Commodore-Amiga-Anwender und Programmierer: eine Entwickler-Dokumentation zum Amiga-Dos-Betriebssystem, Version 1.2. Programmierung, interne Datenstruktur und Diskettenhandling.
Bestell-Nr. 90465
ISBN 3-89090-465-3
DM 59,–/sFr 54,30/öS 460,20

Markt&Technik-Produkte erhalten Sie bei Ihrem Buchhändler, in Computerfachgeschäften oder in den Fachabteilungen der Warenhäuser.

Irrtümer und Änderungen vorbehalten.



Zeitschriften · Bücher
Software · Schulung

Markt&Technik Verlag AG, Buchverlag, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon (089) 4613-0.
SCHWEIZ: Markt&Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Telefon (042) 415656,
ÖSTERREICH: Rudolf Lechner&Sohn, Heizwerkstraße 10, A-1232 Wien, Telefon (0222) 677526, Ueberreuter Media Handels- und Verlagsges. mbH (Großhandel), Alser Straße 24, A-1091 Wien, Telefon (0222) 481538-0



Fragen Sie bei Ihrem Buchhändler nach unserem kostenlosen Gesamtverzeichnis mit über 200 aktuellen Computerbüchern und Software.

Preiswerter geht's kaum

Preisgünstige Spiele für den Amiga sind rar. Meistens muß man schon einiges hinblättern, um zu einem vernünftigen Programm zu kommen. Daß es auch billiger geht, beweist neue deutsche Software.

Unmengen von Münzen verschiedener Nationalitäten sind in die Geldschlitze der »Pac Man«-Automaten verschwunden. Wenn Sie das Spiel mögen und Ihr Geldbeutel einen ausgedehnten Besuch in der Spielhalle nicht zuläßt, sind Sie ein Fall für »Cruncher Factory«.

Die Idee ist zwar schon etwas betagt, aber immer noch reizvoll: Sie steuern eine mampfende Spielfigur durch ein Labyrinth, das von vier Geistern bewacht wird. Ihr Sprite muß alle Punkte auffressen, was die Geister natürlich zu verhindern suchen. Wenn Sie mit einem kollidieren sollten, verlieren Sie ein Leben. Damit Sie nicht ganz wehrlos dastehen, sind über das Spielfeld ein paar Kraftpillen verteilt. Sobald eine solche gefressen wurde, können Sie den Geistern zu Leibe rücken. Dafür winkt Ihnen ein dicker Bonus.

Cruncher Factory hat ein ansprechendes Titelbild und eine gute Grafik, die aber für den Amiga nichts Weltbewegendes ist. Die Soundeffekte sind spärlich und leider teilweise richtiggehend nervend. Erfreulicher ist da schon der Editor, mit dem Sie sich Ihre ganz privaten Labyrinthe zusammenstellen können. Er ist kinderleicht zu bedienen: Mit dem Joystick das gewünschte Symbol anklicken, auf das Spielfeld fahren und draufloskonstruieren. Bis zu 100 Level können Sie nach eigenen Wünschen zusammenstellen.

Cruncher Factory muß man nicht alleine spielen. Sie können auch zu zweit den Screen abräumen — miteinander oder auch gegeneinander. Die besten Spielergebnisse werden auf der Diskette gespeichert. Schade ist nur, daß die »Intelligenz« der Geister nicht sonderlich hoch ist. Dadurch wird das Spiel manchmal fast zu leicht.



Bild 1. Space Battle glänzt mit einem hervorragenden Titel

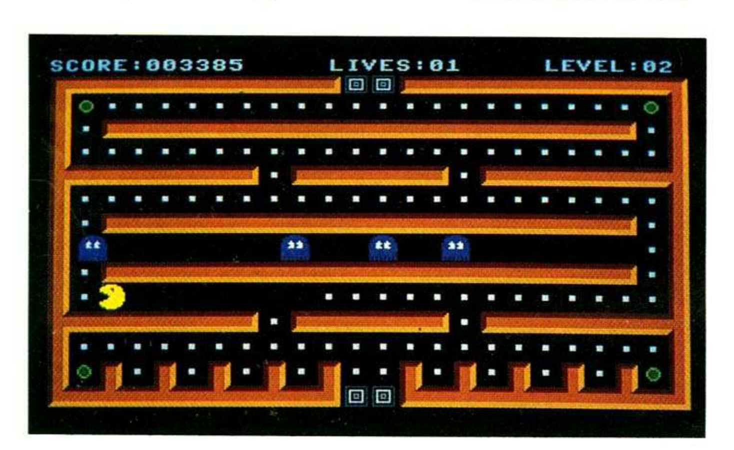
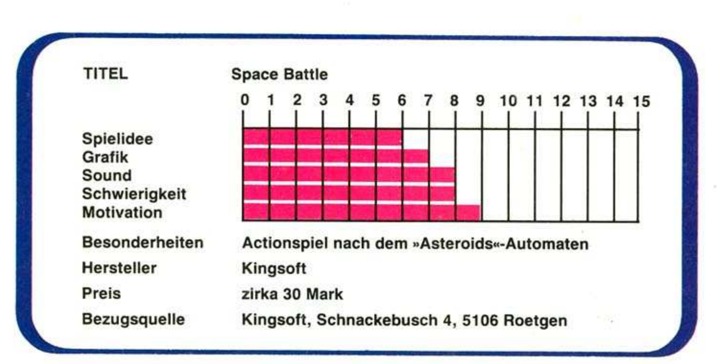


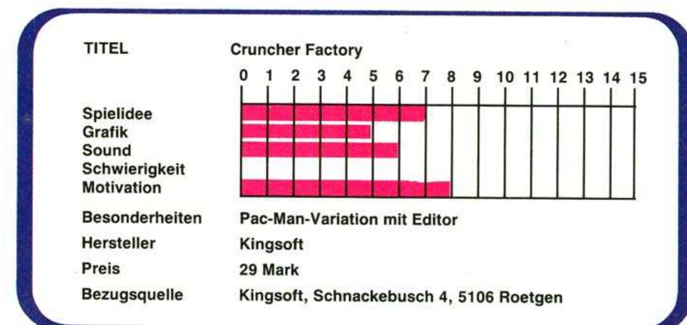
Bild 2. Im Labyrinth der Cruncher Factory

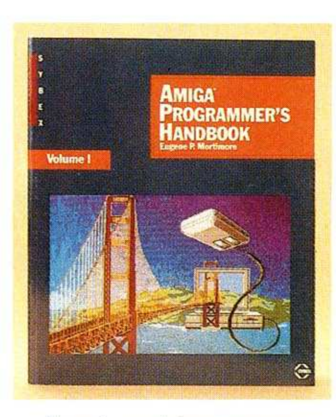
Schneller reagieren als bei Cruncher Factory müssen Sie bei »Space Battle«. Auch diesem Programm liegt ein Spielautomat zugrunde: »Asteroids«. Mit einem Raumschiff und einem High-Zapp-Laser ausgerüstet, säubern Sie die Galaxis von herumfliegenden Meteoren. Wenn Sie einen getroffen haben, zerspringt er in viele kleinere Teile, die Ihnen solange um die Ohren fliegen, bis Sie auch diese abräumen. Wenn Sie mit einem solchen galaktischen Brocken kollidieren, erleiden Sie das gleiche Schicksal: Sie zerschellen. Ihr Schiff ist nach allen Seiten hin beweglich und ist sogar hyperraumtauglich. Wenn akute Gefahr droht, können Sie sich damit blitzschnell aus dem Staub machen und tauchen einige Sekunden später in einer anderen Ecke des Bildschirms wieder auf. Das ist aber mit einer Gefahr verbunden, denn hinterhältigerweise setzt der Bordcomputer Ihr Schiff manchmal auf einen Asteroiden. Damit aber das Ganze so richtig hektisch wird, greifen Sie von Zeit zu Zeit einige fliegende Untertassen an, die teilweise recht gezielt auf Sie schießen. Um hier zu überleben, gibt es also nur eine Devise: ballern, ballern und noch mal ballern. Haben Sie einen Teil der Galaxis von den Brocken befreit, geht's ab zum nächsten. Und hier erwarten Sie schon einige Asteroiden mehr als vorher.

Space Battle ist nicht minder interessant als Cruncher Factory. Auch hier gibt es einen Zwei-Spieler-Modus, der einiges zum Spielwitz beiträgt. Sie können miteinander oder auch gegeneinander antreten, wobei Sie Ihren Gegner gnadenlos vom Feld räumen können, um alleine die Punkte zu sammeln und die High-Score-Liste zu bevölkern. Grafisch ist außer einem guten Titelbild nichts Überragendes geboten, dafür gibt es einige wundervolle Soundeffekte. Space Battle ist ein solide gemachtes Spiel, das trotz der alten Spielidee viel Spaß bringt.

Wenn man bedenkt, daß sowohl Cruncher Factory als auch Space Battle nur knapp 30 Mark kosten, dann fällt einem die Entscheidung zum Kauf nicht mehr schwer. Obwohl die Ideen beider Programme inzwischen einige Jahre auf dem Buckel haben, sind Sie dank einiger Extras wie den Editor oder die speicherbaren High-Score-Listen aufgewertet. Beide haben, obwohl sie billig sind, mehr zu bieten als ähnliche Spiele. (al)







Amiga Programmer's Handbook

Der Amiga mit seinem in C geschriebenen Betriebssystem bietet viele Funktionen an, das Schwierige ist die Funktionsnamen und -aufrufe zu finden. Eine fantastische Hilfe bietet hierbei das Buch »Amiga Programmer's Handbook«.

Zwei Voraussetzungen zur sinnvollen Nutzung des Buches müssen jedoch gegeben sein, zum ersten muß der Leser in der Programmiersprache C schon einiges an Grundwissen mitbringen und zum zweiten muß er zumindest Schulenglisch beherrschen.

Nach einer relativ kurzen Einführung, in der auch sinnvollerweise die verwendete Notation erklärt wird, wird dann mit der Erläuterung der einzelnen Funktionen begonnen. Besonders interessant ist dabei, daß alle Funktionen der Version 1.2 enthalten sind. Die vierzig Funktionen, die seit der Vorgängerversion hinzugekommen sind, werden gekennzeichnet, so daß man immer sofort weiß, ob die Funktion auch mit der Version 1.1 verwendet werden darf.

Die Erklärungen zu den einzelnen Funktionen beginnen immer mit der Syntax des Aufrufs, das genügt bei geübten Programmierern meist, um die Routine zu benutzen. Interessant für Assembler-Programmierer ist, daß bei der Syntax auch steht, welcher Wert in welches Register geladen wird. Damit kann auch mit Assembler auf die Funktionen zurückgegriffen werden.

Nach der Erläuterung, was die Funktion eigentlich tut, werden die einzelnen Parameter mit Beschreibung aufgeführt. Der für Anfänger wichtigste Teil folgt dann in der Form einer ausführlichen Erklärung zu der besprochenen Funktion. Hier werden auch die anderen, in diesem Zusammenhang wichtigen Begriffe, erläutert. An den Stellen, wo es sinnvoll erscheint, werden die Sachverhalte durch Grafiken und Tabellen noch verdeutlicht.

Im Anhang findet man dann noch die wichtigsten Fachbegriffe mit kurzen Beschreibungen, was zum Verständnis des Buches viel beiträgt. Außerdem werden die verschiedenen Grafikmöglichkeiten des Amiga und die notwendigen Schritte zu ihrem Aufbau aufgezeigt. Abgerundet werden sie durch kleine Programmbeispiele zur Erzeugung der verschiedenen Modi.

Insgesamt gesehen ist das Buch für jeden, der den Amiga programmieren möchte, sehr nützlich. Ein wichtiger Punkt dabei ist, daß auch die neuen Funktionen der Version 1.2 enthalten sind. (rb)

Eugene P. Mortimore, Amiga Programmer's Handbook, Sybex Verlag, 635 Seiten, ISBN 0-89588-367-8, Preis 85 Mark



Maschinen- und Assemblersprache des M68000

Obwohl der Titel des Buches den Eindruck erweckt, daß es sich um eine allgemeine Abhandlung des Themas handelt, geht es um drei Computer. Die Programme, die vorgestellt werden, laufen auf dem Sinclair QL, dem NDR-Kleincomputer und dem Amiga.

Gleich am Anfang wird in einem Listing ein kleiner Maschinensprache-Monitor in Basic vorgestellt, mit dem dann weitergearbeitet wird. Überhaupt werden im ersten Teil des Buches immer Basic-Hauptprogramme verwendet, um die Maschinenspracheteile aufzurufen.

Nachdem die Datenformate und der Adreßbus kurz vorgestellt wurden, geht es dann schon zu den Adressierungsarten des M68000. Wie auch später bei allen vorgestellten Problemen, findet man hier kurze Programme, mit denen der Sachverhalt weiter vertieft wird. Allerdings gibt es nicht zu jedem Problem ein Programm speziell für den Amiga, es ist jedoch möglich, die meisten Programme umzusetzen.

Im folgenden Teil des Buches werden Befehle für Schleifen genauso behandelt wie die bedingten und unbedingten Sprünge. Anhand von recht interessanten Beispielen, zum Beispiel Sieb des Eratosthenes und den Türmen von Hanoi, werden wichtige Fakten aufgezeigt.

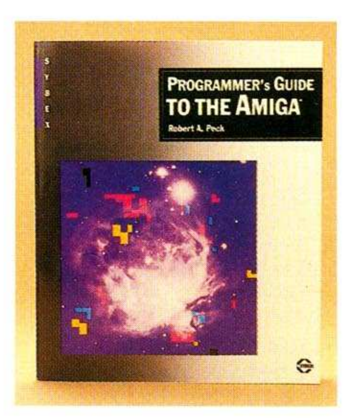
Danach wird für jeden der drei Computer ein Assembler kurz vorgestellt.

Mit den Befehlen für Multiplikation und Division von Fließkommazahlen, die anschließend vorgestellt werden, kann sicher jeder etwas anfangen. Das folgende Programm dreht sich um die Türme von Hanoi, da das Listing aber auch die grafische Darstellung beinhaltet und für den NDR-Kleincomputer geschrieben ist, hat man als Amiga-Besitzer leider das Nachsehen.

Nur für Fortgeschrittene gedacht ist der vollständig abgedruckte Disassembler für den M68000. Einzelne Teile des Programmes muß der Leser nämlich selbst entwickeln, als Beispiel sei hier die Ausgabe auf dem Bildschirm genannt.

Das Buch eignet sich nur für denjenigen, der das Handbuch und andere Literatur zu seinem Computer schon gelesen und verstanden hat. Das ist auch nicht verwunderlich, da man auf 363 Seiten nicht alle Möglichkeiten eines Computers beschreiben kann. Vielleicht wäre es aber sinnvoller gewesen, sich trotzdem auf einen Computer zu beschränken. (rb)

Helmut Ostermann, Maschinen- und Assemblersprache des M68000, Vogel Buchverlag, 363 Seiten, Preis 48 Mark



Programmer's Guide to the Amiga

Nach einer kurzen Einführung, in der einige Grundlagen erwähnt werden, folgt ein Teil, der sich mit der Programmierung des Amiga-DOS beschäftigt. Genau wie im restlichen Buch muß man allerdings zwei Sprachen beherrschen, nämlich Englisch und C. Angesprochen werden in diesem Teil so interessante Themen wie Arbeiten mit Dateien und Ansprechen von Druckern. Die Benutzung von Amiga-DOS-Befehlen wie DIR wird anhand von kurzen Programmen erklärt. Für viele Anwendungen ist es wichtig, daß verschiedene Directories verwendet werden, was sehr ausführlich erläutert wird.

Nach einem Ausflug in das Multitasking, der nur für versierte Programmierer gedacht ist, kommt ein Kapitel über Grafik. Hier findet man alles Wissenswerte über alle Funktionen, die auf dem Amiga verfügbar sind. Die verschiedenen Refresh-Methoden für Fenster werden genauso behandelt wie die Menütechnik.

Das folgende relativ kurze Kapitel beschäftigt sich mit den sogenannten Devices, wobei auch auf den Timer eingegangen wird, der oft nützlich sein kann. Für denjenigen, der auf dem Amiga bewegte Grafik erzeugen will, ist der nächste Abschnitt gedacht. Hier geht es in der Hauptsache um Sprites und Bobs, wobei auch die Vor- und Nachteile der virtuellen Sprites erläutert werden.

Nach dem Teil über die Soundmöglichkeiten des Amiga, kommt dann wieder ein Kapitel für Fortgeschrittene über das Multitasking und dessen Programmierung. Aber auch einfachere Beispiele sind hier zu finden.

Die zwei letzten Abschnitte beschäftigen sich mit dem Texteditor »ED« und dem C-Compiler von Lattice, was für den Programmierer allerdings relativ uninteressant ist.

Insgesamt ist das Buch gut geeignet, Wissen über den Amiga und seine Programmierung zu bekommen. Durch die vielen Beispielprogramme findet man schnell Zugang zu den vorgestellten Funktionen (rb)

Robert A. Peck, Programmer's Guide to the Amiga, Sybex Verlag, 351 Seiten, ISBN 0-89588-310-4, Preis 99 Mark



Gewinnen Sie!
2000 Mark . . .
Gewinnen Sie!
■ ■ ■ ist uns Ihr Programm wert, wenn es als Programm des Monats im Amiga-Magazin veröffentlicht wird. — Haben Sie für Ihren Amiga ein Super-Programm geschrieben?
Dann gibt es nur eines: einschicken an die Redaktion des Amiga-Magazins. Wir wählen aus den besten Listings, die wir veröffentlichen, ein Programm des Monats aus, das mit 2000 Mark honoriert wird.
Für Sie also eine Mühe, die sich lohnt. Ob Sie nun ein Action-Spiel oder eine ernsthafte Anwendung auf Lager haben, gute Programme sind immer willkommen.
Haben Sie sinnvolle und mächtige Betriebssystem-Erweiterungen programmiert, die die Bedienung des Amiga vereinfachen (Diskmonitore, CLI-Erweiterungen), hier haben Sie die Chance, anderen Ihre Entwicklung zugute kommen zu lassen und auch noch etwas dabei zu verdienen.
Schicken Sie Ihr Programm an:
Markt & Technik Verlag AG
AMIGA-Redaktion
Stichwort: Programm des Monats
Hans-Pinsel-Str. 2, 8013 Haar bei München
Gewinnen Sie!
Gewinnen Sie!

Ihr Programm im Amiga-Magazin

Haben Sie bereits ein interessantes Programm für den Amiga geschrieben? Dann sollten Sie sich vielleicht überlegen, ob das von Ihnen entworfene Programm nicht für die Allgemeinheit der Amiga-Besitzer nützlich ist.

Nicht nur neue Spiele werden durch neue und leistungsfähige Computer wie den Amiga zu faszinierenden Meilensteinen in der Heimcomputer-Geschichte. Durch die steigende Speicherkapazität und die immer professionelleren Betriebssysteme können auch immer mehr höhere Programmiersprachen eingesetzt werden. Da die einzelnen Systeme heute immer komplexer aufgebaut sind, werden sogar die Betriebssysteme selbst in einer höheren Sprache erstellt. Auch hier ist der Amiga wieder ein sehr gutes Beispiel. Seine Seele, das Betriebssystem, wurde mit der modernen Sprache C programmiert.

Da für den Amiga in der Zwischenzeit ein bunt gemischtes Sammelsurium an verschiedensten Sprachen zur Verfügung steht, von Fortran bis Prolog, versorgen wir Sie natürlich mit brandheißen Informationen zu diesen Sprachen. Dabei wollen wir natürlich nicht an unseren Leser »vorbei programmieren«. Vielmehr haben Sie hier die Möglichkeit, Ihr Amiga-Magazin als Leser mitzugestalten. Wir suchen vor allem kurze Programme mit großer Wirkung, also Utilities oder wirkungsvolle Routinen, die jeder Leser und Programmierer sehr leicht für sich aus- und verwerten kann. Besonders nützlich ist auch eine genaue Dokumentierung des Quellcodes. Dadurch erschließt sich die von Ihnen verwendete Programmtechnik auch allen anderen Amiga-Anwendern. Die Programme werden leicht nachvollziehbar, und bestimmte Teile können so auf einfache Weise in eigene Programme eingebaut werden. So steigert sich ganz allmählich auch die Qualität der veröffentlichten Programme. Natürlich wird Ihr Beitrag bei Veröffentlichung entsprechend honoriert. Welche Sprache oder welchen Compiler Sie dabei verwenden, ist vollkommen egal. Ob Sie nun das bereits vorhandene Amiga-Basic benutzen oder sich mit Maschinensprache und C herumschlagen, wir werden Ihre Einsendung bewerten und, wenn für gut befunden, in einer der folgenden Ausgaben veröffentlichen. Wie gesagt, sie können jede Sprache verwenden, die für den Amiga erhältlich ist. Für diese Programme wird es dann eine eigene Rubrik geben, in der Sie die besten Lösungen gesammelt nach den einzelnen Sprachen geordnet wiederfinden.

Helfen Sie mit, Ihr Amiga-Magazin informativ und anregend zu gestalten. Sie haben die Möglichkeit dazu.

Wie gehen Sie bei der Einsendung vor? Ganz einfach, schicken Sie uns Ihr Programm mit genauer Beschreibung und Diskette, sowie der Angabe der verwendeten Sprache an folgende Adresse:

Markt & Technik Verlag AG
Amiga-Redaktion
z. H. Herrn R. Fieger
Stichwort:
Amiga-Programmierung
Hans-Pinsel-Str. 2c
8013 Haar bei München

Gesucht: Tips & Tricks zum Amiga

Sie sind ein begeisterter Amiga-Fan? Sie kennen Ihren Computer inzwischen so gut, daß die Begriffe »CLI«, »Amiga-DOS« oder »Workbench« für Sie keine Fremdwörter mehr darstellen? Dann beteiligen Sie sich doch an der Tips & Tricks-Rubrik »Ihres« neuen Magazins!

Die Tips & Tricks-Ecke im Amiga-Magazin ist eine reine »von-Lesern-für-Leser«-Rubrik. Das bedeutet, daß Sie durch Ihre Beiträge und durch Ihre guten Ideen anderen Lesern helfen können, mit dem neuen Computer besser zurechtzukommen. Sicherlich erinnern Sie sich noch an die Zeit, wo Sie begierig jede Zeile Information zum Amiga mit Heißhunger verschlangen. Gerade als Einsteiger ist man auf die Erfahrung und das Wissen von »Profis« angewiesen.

Sie haben eine neue Idee, wie man mit einem kleinen Amiga-DOS-Programm die Arbeit mit dem Computer vereinfachen kann? Einschicken!

Sie haben ein raffiniertes, kurzes Basic-, C- oder Assembler-Programm ausgeklügelt? Einschicken!

Sie haben einen Trick herausgefunden, wie man mit der einen oder anderen Software komfortabler arbeiten kann? Einschicken!

Sie haben Ihren Amiga durch eine kleine Bastelei aufgewertet? Einschicken! Jede gute Idee wird in »Ihrer« Rubrik veröffentlicht. Natürlich bekommen Sie dann von uns ein entsprechendes Honorar. Wichtig ist nur, daß Sie in einer kurzen Beschreibung den Grundgedanken Ihrer Idee zu Papier bringen. Wenn Sie ein Programm einschicken, sollten Sie nicht vergessen, eine Diskette mitzuschicken. Und sollte einmal ein Beitrag, aus welchen Gründen auch immer, nicht zur Veröffentlichung geeignet sein, bekommen Sie Ihre gesamten Unterlagen natürlich wieder zurückgesandt.

Schreiben Sie an:
Markt & Technik Verlag AG
Redaktion Amiga-Magazin,
Aktion Tips & Tricks
Hans-Pinsel-Str. 2
8013 Haar bei München

Eines sollten Sie nicht vergessen: Sie müssen kein Vollprofi sein, um bei dieser Aktion mitzumachen. Auch Tricks, die für Sie schon längst zur Routine geworden sind, könnten anderen Lesern eine wichtige Hilfe sein. Vor allem kurze und leicht durchschaubare Listings sind begehrt. Wir wünschen Ihnen viel Spaß beim Programmieren und Knobeln! (tr)



ELECTRONIC ARTS

ELECTRONIC ARTS DELUXE-Serie

Exklusivvertrieb bei Markt&Technik

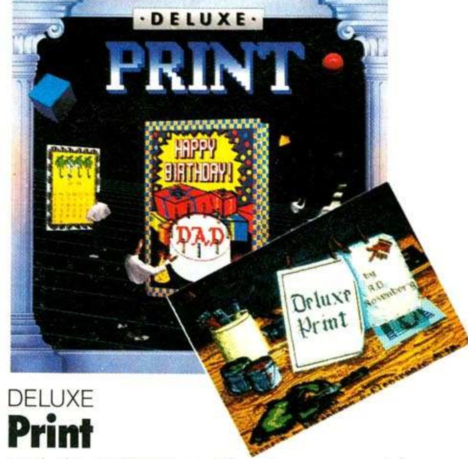


NEU DELUXE

Paint II

Deluxe Paint II ist eines der außergewöhnlichsten Grafikprogramme auf dem Softwaremarkt. Im Vergleich zu seinem Vorgänger ist Deluxe Paint II in der Lage, in der Textfunktion Fett- und Kursivschrift sowie Unterstreichungen einzubauen, perfekte Manipulation in der dritten Dimension darzustellen oder wichtige Parameter wie Längsdruck, Schwarzweißgraustufen - oder Farbdruck, Ränder und vieles mehr unmittelbar vor dem Druck per Mausklick zu verändern.
Hardware-Anforderungen:
Amiga 512 Kbyte und Farbmonitor
Bestell-Nr. 52568
DM 299,–* (sFr 249,–/öS 2990,–)
DELUXE **Paint I**
Hardware-Anforderung:
Amiga 256 Kbyte und Farbmonitor
Bestell-Nr. 52565
Nur noch DM 199,–* (sFr 199,–)

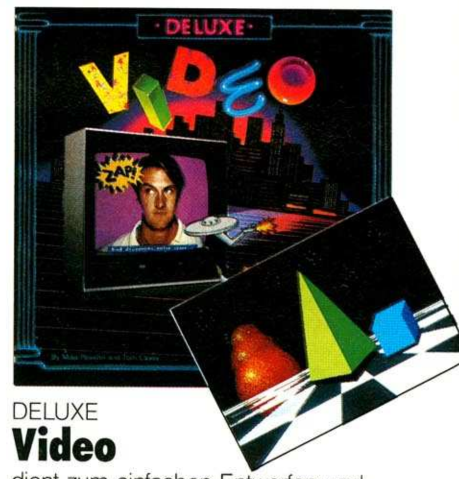
In Vorbereitung:
Music Construction Set
Das professionelle Musikprogramm für Ihren Amiga!
Hardware-Anforderung: Amiga 512 Kbyte.
Bestell-Nr. 52569
DM 249,–* (sFr 199,–/öS 2290,–*)



DELUXE

Print

und ein grafikfähiger Drucker verwandeln den Amiga in eine Druckmaschine. Sie können Karten, Poster, Briefköpfe und vieles mehr auf einfachste Weise entwerfen und ausdrucken. Besitzer eines Farbdruckers können ihr Werk auch in Farbe aufs Papier bringen. »Deluxe Print« ist kompatibel zu »Deluxe Paint«. Das bedeutet, daß man Grafiken zwischen den Programmen austauschen kann.
Hardware-Anforderungen:
Amiga (512 Kbyte) und Farbmonitor.
Bestell-Nr. 52566
DM 249,–* (sFr 199,–/öS 2290,–*)



DELUXE

Video

dient zum einfachen Entwerfen und Zusammenstellen von animierten Grafik-Sequenzen. Sie können so Videofilme mit Computergrafik versehen und regelrechte Computer-Videoclips zusammenstellen.
Das Programm ist ebenfalls kompatibel zu »DELUXE PAINT« und »DELUXE PRINT«.
Hardware-Anforderungen:
Amiga (512 Kbyte) und Farbmonitor.
Bestell-Nr. 52567
DM 249,–* (sFr 199,–/öS 2290,–*)
*inkl. MwSt. Unverbindliche Preisempfehlung

Die ideale Ergänzung:
Das Buch zur Software
Markus Breuer
DELUXE Grafik mit dem Amiga
1986, 370 Seiten
ISBN 3-89090-412-2
Best.-Nr. 90412
DM 49,– (sFr 45,10/öS 382,20)
Erhältlich bei Ihrem Buchhändler



Markt&Technik
Zeitschriften · Bücher
Software · Schulung

Deluxe Paint, Deluxe Print und Deluxe Video erhalten Sie in den Fachabteilungen der Kaufhäuser, in Computershops, bei Commodore-Fachhändlern oder direkt beim Verlag gegen Vorauskasse.

Markt&Technik Verlag AG, Buchverlag, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon (089) 4613-0

Bestellungen im Ausland bitte an: SCHWEIZ: Markt&Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Telefon (042) 415656 · ÖSTERREICH: Rudolf Lechner & Sohn, Heizwerkstraße 10, A-1232 Wien, Telefon (0222) 677526 · Ueberreuter Media Verlagsges. mbH (Großhandel), Alser Straße 24, A-1091 Wien, Telefon (0222) 481538-0.

703228

DIE NÄCHSTE AUSGABE ERSCHEINT AM 30.9.

Vorschau 10/1987



Spielen, was sonst?

Wegen seiner hervorragenden Grafik und dem fantastischen Sound ist der Amiga für viele ein attraktiver Spielecomputer. Deswegen wird es in der nächsten Ausgabe Programmtests aus zwei Spielesparten geben, die recht unterschiedlich sind, die aber immer wieder ihre Fans finden: Sport- und Strategiespiele. Bei den Sportspielen geht es im wesentlichen um Golfsimulationen, während Strategiespiele mit geopolitischem Ausmaß für heute und die Zukunft etwas näher unter die Lupe genommen werden.

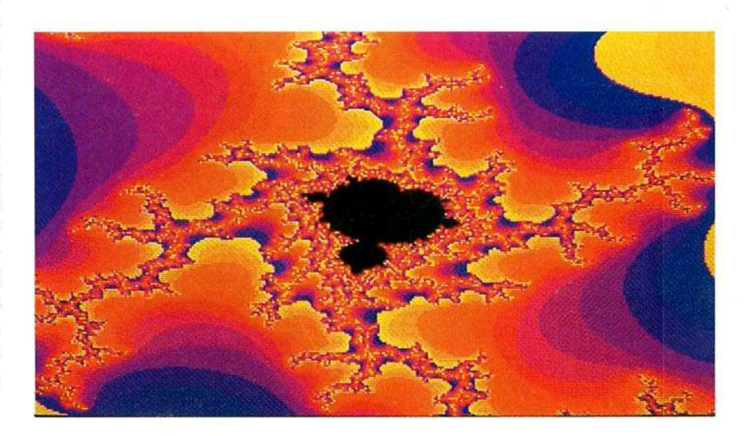
Kein Handwerker ohne Werkzeug

Utilities braucht jeder, der vernünftig mit dem Amiga arbeiten will. In unserem Listingteil werden wir daher wieder einige nützliche Programme zum Abtippen veröffentlichen. Zusätzlich gibt es natürlich wieder entsprechende Softwaretests: Der Boot-Boy für das Erstellen eigener Vorspänne oder »Power Windows«, ein unerläßliches Tool für C-Programmierer. Für alle, die sich mit Musik auf dem Amiga beschäftigen, wird sicher unser Vergleich von Sonix und dem Deluxe Music Construction Set interessant sein.

Grafik und Monitore

In der nächsten Ausgabe des Amiga-Magazins werden wir uns ausführlich mit der Grafik auf dem Amiga beschäftigen. Für Softwaretests sind gute Programme aus diesem Bereich vorgesehen. Neben dem Grafik-Tool »Grabbit«, das man dazu benutzen kann, jegliche Grafik, die sich auf dem Bildschirm des Amiga zeigt, abzugreifen, gibt es eine ausführliche Vorstellung des neuen »Butcher«. Dieses fantastische Utility-Programm ermöglicht das Nachbearbeiten von IFF-Bildern auf eine Art und Weise, bei der selbst Deluxe-Paint die Flaggen streichen muß. Natürlich wird sich dieser Schwerpunkt auch in den Listings wiederfinden. Bunte Programme und Spiele mit Grafik sind gefragt. Die Hardware zur Eingabe und Bearbeitung von Grafik soll auch nicht vergessen werden. So finden Sie im nächsten Heft zum Beispiel einen Test des Digitalisierers »Digi View«.

Als weiterer Schwerpunkt ganz anderer Art sind Monitore geplant. Wir befassen uns mit dem Anschluß des Commodore 1901-Monitors über den RGB-Port Amiga. Außerdem werden die PAL-Modulatoren näher untersucht und verschiedene Monitore getestet.



Inserenten

Inserent	Seite
Amigaland	35
Atlantis Soft	113, 114
Bellingrath	79
Bittendorf	113
CAS	79
Commodore	5, 133
Compu Store	66
CSJ Computersoft	47
Data Becker	18/19, 31
DTM	123
Eco Soft	33
Epson	2
EZ Appel & Grywatz	75
H-Soft	33
Island Software	57
Jumbo Soft	105
Juris	66
Kingsoft	15
Kröning	47
Kupke	22
Markt & Technik Buchverlag	44, 73, 108, 125, 130
Ossowski	75
PD-Shop	113
Philgerma	63
Rat + Tat	47
SAS Bernd	51
Schmielewski	66
Softwareland	39, 134
Soyka Datentechnik	52
Stalter	75
Stutz	66
SWS	79
Tröps	51
Vesalia Versand	66
Viza Software	123

Impressum

Herausgeber: Carl-Franz von Quadt, Otmar Weber

Geschäftsführender Chefredakteur: Michael Scharfenberger

Chefredakteur: Albert Absmeier (aa)
Stellvertr. Chefredakteur: Georg Klinge (gk)
Redaktion: Peter Aurich (pa), René Beaupoil (rb), Jörg Kähler (jk), Dieter Mayer (dm), Arnd Wängler (aw)

Hotline: Gerd Donaubauer (640), Monika Welzel (640)

Redaktionsassistenz: Monika Lewandowski (222), Andrea Kaltenhauser (202), Bärbel Pasternok (202)
Fotografie: Janos Feitser/Jens Jancke

Layout: Leo Eder (Ltg.), Rolf Raß (Cheflayouter), Dagmar Berninger, Willi Gründl

Auslandsrepräsentation:
Schweiz: Markt & Technik Vertriebs AG, Kollerstr. 3, CH-6300 Zug, Tel. 042-415656, Telex: 862329 mut ch
USA: M & T Publishing, Inc: 501 Galveston Drive, Redwood City, CA 94063; Tel. (415) 366-3600, Telex 752-351

Manuskripteinsendungen: Manuskripte und Programmlistings werden gerne von der Redaktion angenommen. Sie müssen frei sein von Rechten Dritter. Sollten sie auch an anderer Stelle zur Veröffentlichung oder gewerblichen Nutzung angeboten werden, so muß dies angegeben werden. Mit der Einsendung von Manuskripten und Listings gibt der Verfasser die Zustimmung zum Abdruck in von der Markt & Technik Verlag AG herausgegebenen Publikationen und zur Vervielfältigung der Programmlistings auf Datenträger. Mit der Einsendung von Bauanleitungen gibt der Einsender die Zustimmung zum Abdruck in von Markt & Technik Verlag AG verlegten Publikationen und dazu, daß Markt & Technik Verlag Geräte und Bauteile nach der Bauanleitung herstellen läßt und vertreibt oder durch Dritte vertreiben läßt. Honorare nach Vereinbarung. Für unverlangt eingesandte Manuskripte und Listings wird keine Haftung übernommen.

Produktionsleitung: Klaus Buck

Anzeigenverkaufsleitung: Ralph Peter Rauchfuss (126)

Anzeigenleitung: Alicia Clees (313)

Anzeigenverwaltung und Disposition: Patricia Schiede (172), Monika Burseg (147)

Anzeigenformate: 1/1-Seite ist 266 Millimeter hoch und 185 Millimeter breit (3 Spalten à 58 mm oder 4 Spalten à 43 Millimeter). Vollformat 297 x 210 Millimeter. Beilagen und Beihefter siehe Anzeigenpreisliste.

Vertriebsleiter: Helmut Grünfeldt (189)

Vertrieb Handelsauflage: Inland (Groß-, Einzel- und Bahnhofsbuchhandel) sowie Österreich und Schweiz: Pegasus Buch- und Zeitschriften-Vertriebsgesellschaft mbH, Hauptstätterstraße 96, 7000 Stuttgart 1, Telefon (0711) 6483-0

Bezugsmöglichkeiten: Leser-Service: Telefon 089/4613-249. Bestellungen nimmt der Verlag oder jede Buchhandlung entgegen. Das Abonnement kann jederzeit zum Ende des bezahlten Zeitraums gekündigt werden.

Bezugspreise: Das Einzelheft kostet DM 7,—. Der Abonnementspreis beträgt im Inland DM 79,— pro Jahr für 12 Ausgaben. Darin enthalten sind die gesetzliche Mehrwertsteuer und die Zustellgebühren. Der Abonnementspreis erhöht sich um DM 18,— für die Zustellung im Ausland, für die Luftpostzustellung in Ländergruppe 1 (z. B. USA) um DM 38,—, in Ländergruppe 2 (z. B. Hongkong) um DM 50,—, in Ländergruppe 3 (z. B. Australien) um DM 68,—.

Druck: R. Oldenbourg GmbH, Hürderstr. 4, 8011 Kirchheim

Verantwortlich: Für redaktionellen Teil: Albert Absmeier. Für Anzeigen: Alicia Clees.

Redaktions-Direktor: Michael M. Pauly

Vorstand: Carl-Franz von Quadt, Otmar Weber

Anschrift für Verlag, Redaktion, Vertrieb, Anzeigenverwaltung und alle Verantwortlichen:
Markt & Technik Verlag Aktiengesellschaft, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon 089/4613-0, Telex 522052

Telefon-Durchwahl im Verlag:

Wählen Sie direkt: Per Durchwahl erreichen Sie alle Abteilungen direkt. Sie wählen 089-4613 und dann die Nummer, die in Klammern hinter dem jeweiligen Namen angegeben ist.

MIT AMIGA KREATIV IN DIE ZUKUNFT



Das Amiga-Computermagazin eröffnet Ihnen alle Möglichkeiten, die in Ihrem Amiga stecken und ihn noch kreativer machen. Es bietet Ihnen Hinweise zur Programmierung, tolle Listings, die neuesten Hard- und Softwaretests und natürlich alle Neuigkeiten zum Thema Amiga.

Wir liefern Ihnen die Amiga-Faszination jeden Monat direkt ins Haus.

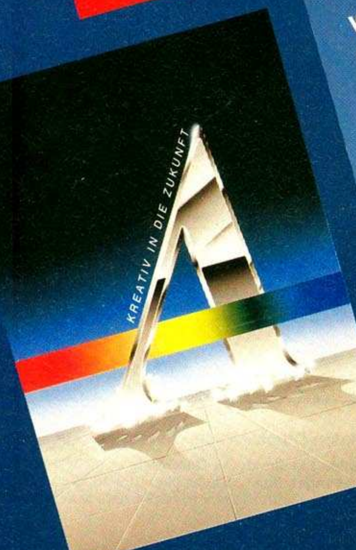
Füllen Sie dafür einfach die nebenstehende Bestellkarte für ein persönliches Abonnement aus und nutzen Sie die Vorteile eines Jahresabonnements:

- pünktliche und regelmäßige Lieferung frei Haus
- und den attraktiven Preisvorteil: Sie erhalten 12 Hefte für DM 79,– statt DM 84,– im Einzelverkauf.

Verschenken Sie doch einfach jede Menge Amiga-Kreativität! Das Amiga-Magazin überrascht jeden Amiga-Fan mit der ganzen Faszination der einzigartigen Möglichkeiten dieses Computers.

Selbstverständlich können Sie sich das Amiga-Magazin auch schenken lassen. Dafür brauchen Sie nur die nebenstehende Bestellkarte für ein Geschenkabonnement ausfüllen. Auf Wunsch gibt es außerdem eine Geschenkurkunde.

DAS AMIGA BEGRÜSSUNGS POSTER



Wenn Sie dieses Abonnement-Angebot nutzen, erhalten Sie mit der ersten Ausgabe das abgebildete Amiga-Poster im Originalformat (84×60 cm) gratis.

Dieses Poster gehört in jedem Fall Ihnen – auch dann, wenn Sie Ihre Bestellung widerrufen.

GO AMIGA! Datei

DM 199.-

Die professionelle Dateiverwaltung!
Spezielle Serie von Datendisketten zusätzlich erhältlich

- *Bild + Tonverwaltung*
- *Listenmodus*
- *Formularmodus*
- *32000 Datensätze pro Datei*
- *32 Felder pro Datensatz*
- *10 Feldtypen*
- *unterstützt Umlaute*
- *Schnelles Sortieren + Suchen*
- *Multitasking*
- *unterstützt alle RAM*
- *kompatibel zu Harddisk*
- *Handbuch deutsch*

GoAmiga! Datei

Mitgliederverwaltung

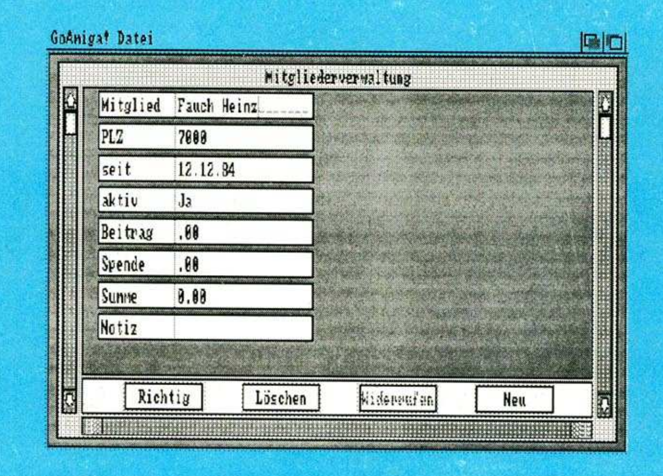
	Mitglied	PLZ	seit	aktiv	Beitrag	Spende	Summe
1	Fauch Heinz	7000	12.12.84	Ja	.00 DM	.00 DM	0.00 DM
2	Klein Martin	8050	1.1.85	Ja	50.00 DM	100.00 DM	150.00 DM
3	Jaspo Karl	5000	3.3.85	Ja	50.00 DM	250.00 DM	300.00 DM
4	Hugi Karl	3400	4.11.85	Nein	100.00 DM	.00 DM	100.00 DM
5	Rappo David	3560	3.12.85	Nein	100.00 DM	250.00 DM	350.00 DM
6	Dorf Emil	2350	12.4.86	Ja	50.00 DM	20.00 DM	70.00 DM
7	Commi Raspo	4500	9.11.86	Ja	50.00 DM	50.00 DM	100.00 DM
8	Sommer Uli	3670	3.12.86	Nein	100.00 DM	350.00 DM	450.00 DM
9	Hugi Hans	9000	11.12.86	Nein	100.00 DM	.00 DM	100.00 DM
10	Gugg Josef	4000	1.1.87	Ja	50.00 DM	350.00 DM	400.00 DM
11	Kloos Peter	3000	1.1.87	Nein	100.00 DM	500.00 DM	600.00 DM
12	Meir Hans	3400	11.1.87	Ja	50.00 DM	250.00 DM	300.00 DM
13	Ottlitz Karl	1000	3.3.87	Ja	50.00 DM	50.00 DM	100.00 DM

Listenmodus

- Einfacher Überblick aller Daten
- Freie Definition von Spaltenbreite, Anordnung und Spaltenbündigkeit (links-rechtsbündig, zentriert)
- Beträge anzeigen mit internationalen Währungen
- Schnelles Wechseln zwischen selektierter Liste und Gesamtliste
- Komfortable Druckersteuerung mit verschiedenen Schriften (Pica, Elite, Fine)
- Individueller Druck (Seitenhöhe, Seitenbreite, Kürzungen usw.)
- Automatisches Anwählen einer Telefonnummer

Formularmodus

- Individuelle Formulargestaltung durch einfaches Feldverschieben mit der Maus
- Variable Feldlänge, auch wenn Daten eingegeben
- Optionales Großschreiben des ersten Buchstabens
- Unabhängiges Abspeichern zum Listenmodus
- Help Taste
- Etiketten beliebig ausdruckbar (bis zu 16-bahnig)
- Automatische Fehlerkorrektur



Bestellservice:

BRD: 0041-1-3115959
CH: 01-3115959

Geschäftszeiten:
10.00–12.30, 13.30–18.30 Uhr, außer montags,
Sa.: 10.00–16.00 Uhr.

Versand ins Ausland nur Vorkasse (Scheck, bar) zzgl. DM 6,– Porto. Händleranfragen erwünscht.

softwareland
Franklinstraße 27
CH-8050 Zürich (Schweiz)